The Mug-Shot Search Problem: A Study of the Eigenface Metric, Search Strategies, and Interfaces in a System for Searching Facial Image Data

The Mug-Shot Search Problem

A Study of the Eigenface Metric, Search Strategies, and Interfaces in a System for Searching Facial Image Data

Ellen Baker

TR-16-98
December 1998

The Mug-Shot Search Problem

A Study of the Eigenface Metric, Search Strategies, and Interfaces in a System for Searching Facial Image Data

A thesis presented by

Ellen Jill Baker

to

The Division of Engineering and Applied Sciences

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

in the subject of

Computer Science

Harvard University

Cambridge, Massachusetts

January 1999

The Mug-Shot Search Problem

A Study of the Eigenface Metric, Search Strategies, and Interfaces in a System for Searching Facial Image Data

Ellen Jill Baker
Thesis Advisor: Margo Seltzer

Abstract

This thesis presents an investigation of methods for conducting an efficient look-up in a pictorial "phonebook" (i.e., a facial image database). Although research on efficient "mug-shot search" is under way, little has yet been done to evaluate the effectiveness of various proposed techniques, and much work remains before systems as practical or ubiquitous as phonebooks are attainable. The thesis describes a prototype system based on the idea of combining a composite face creation method with a face-recognition technique, so that a user may create a facial image and then automatically locate other similar-looking faces in the database. Several methods for evaluating such a system are presented as well as the results and analysis of a user-study employing the methods.

Three basic system components are considered and evaluated: the metric for determining which faces are most similar in appearance to a given "query" face, the interface for producing the query face, and the search strategy. The data demonstrate that the Eigenface metric is a useful (though imperfect) model of human perception of similarity between faces. The data also show how the lack of agreement among people about which faces are most similar to a query limits what can be reasonably expected from any metric. Via simulation, it is demonstrated that, if indeed there were a single human metric for assessing facial similarity, and if the Eigenface metric correlated perfectly with this human metric, then simple interactive hill-climbing in the space of the database images would be an

excellent search strategy, capable of reducing the average number of image inspections required in a search to about 2% of the database. But this superiority of hill-climbing in principle is not sustained in practice, given the observed level of correlation between the Eigenface similarity metric and the “human” one. The average number of image inspections required for the hill-climbing strategy was, in fact, closer to 35% of the database. While this represents an improvement over the 50% required on average for a simple sequential search of the data, it is still insufficient for practical use. However, given the actual performance of the Eigenface metric, the study data show that a non-iterative strategy of constructing a single query image that is a composite of selected features from 100 random database faces is a better approach, reducing the average number of image inspections to about 20% of the database. These and other examples demonstrate and quantify the benefits of an interface in which the Eigenface metric is combined with a composite creation system.

Acknowledgments

This thesis is dedicated to my family, without whom it could never have been written. My children, Sophia and Olivia, come first, as they always have (which is my main excuse for having taken so long to complete the thesis). The joy they bring to my life permeates everything I do, so they are as much a part of this thesis as anyone. It is my hope that whatever they lost by having a mother who was not home as much as they might have liked is well compensated for by having a happier mom. I got to do what I wanted, which I hope helps teach them that they can, too (as long as it doesn't involve squeezing the cat too tightly or refusing to clean up the tinker toys).

A defining moment of my graduate school career was when Margo Seltzer agreed to be my advisor. Her integrity, dedication to teaching, commitment to her students, drive and inspiration as a scientist, and skill as a manager are awe-inspiring. I did not know these things about her when we started and, six years later, I'm still marveling at my good fortune. She gave me the freedom to find a thesis topic that suited me, tackled it enthusiastically with me, skillfully steered me away from the black holes, and guided me through an invaluable learning experience. She pushed hard sometimes, but never forgot to point out the positive. Of special importance to me was her understanding about the dilemmas of juggling work and family. She provided just the right combination of flexibility, pressure, and support that I needed to get the job done.

The other members of my thesis committee have all made important contributions to my education and this thesis. Every conversation I had with Stuart Shieber about this work left me infused with new and interesting ways to think about it; his influence is present in many parts of the thesis. Sandy Pentland also contributed ideas and generously allowed me to work with the Photobook face database and to use the Eigenface engine developed in his group at MIT. Tom Cheatham was especially supportive during the early part of my graduate career.

I thank my husband and friend Mark Sommer for patiently supporting me in this project and for all the recent solo parenting he did on weekends. Without him as a teammate, I couldn't have started, let alone finished. My father Adolph Baker read drafts with unparalleled care, corrected errors, and made many suggestions that improved my writing (notwithstanding my decision to boldly split my infinitives). An enormous side benefit to the final throes of this project is the fun I have had discussing the work with him. I thank him for his input, insight, and friendship. My mother Dora Baker has also been a constant, loyal, and ever-helpful friend. She pinch-hit whenever I needed her for child care or anything else. Her dual interest in both fine art and science was probably the model for my eclectic studies. My sister Linda Baker and brother Danny Baker also each played an important role. (I think it was Danny who asserted that the reason I needed a PhD. was to keep up with my brother and sister. He might be right, in which case they deserve special thanks for providing the necessary impetus for this thesis.) Thanks also to my other parents, Armin and Connie Sommer, who further broadened the incredible support structure of my family.

I would like to thank Barbara Grosz for her feedback, for all the AI Research Group brunches, for encouraging me to present my work at the AIRG meetings, and for generally trying to foster a sense of community in the department. An encouraging word from her always meant a lot to me. Joe Marks was a great teacher, ever enthusiastic, supportive, and full of interesting ideas. Michael Rabin taught what may be my favorite classes ever; I learned a great deal from him. I would also like to thank Jim Clark for his supportiveness and for the fun class on genetic algorithms.

Henry Leitner taught my very first introductory class in computer science at the Harvard Extension School, as well as several other early classes. His excellent teaching and the Extension School CAS program provided a critical first stepping stone in my path.

Many fellow graduate students offered their help and friendship throughout. I would particularly like to thank Cecile Balkanski, Karen Daniels, Dan Ellard, Cesar Galindo, Josh Goodman, Rebecca Hwa, Luke Hunsberger, Karen Lochbaum, Ted Nesson, Wheeler Ruml, Dan Roth, Kathy Ryall, Nadia Shalaby, Chris Small, Keith Smith, and Dave Sullivan, who each played a significant part in my education. I learned as much from them as I did in classes and it was a lot more fun. Ted and Wheeler, whose work hours intersected most often with mine, deserve extra office-mate thanks for so graciously tolerating my many questions and interruptions to their work.

Heartfelt thanks go to everyone who gave their time to participate in my user study. In this fast-paced world, that contribution was a big one. I considered naming every one of them here, but then realized that would violate my promise of anonymity to study subjects.

I would also like to thank Grace Myhill for her helpful consulting and her interest in my work. Thanks to Wasi Wahid for his help answering technical questions and his assistance with producing the Eigenfeature coefficients.

Thank you falls way short of expressing my appreciation to the people who have taken such great care of my children while I worked. The fabulous teachers at Arlington Children's Center and the Fayerweather Street School, and my friends Lingya, Natasha, Virginia, and Jennifer all deserve special mention.

Finally, I would like to thank Terry Dankel and Anne Berg for their particularly supportive friendships.

Chapter 1
Introduction

1.1 The Mug-Shot Search Problem

Suppose you have an interesting conversation with someone at a conference or seminar, but the conversation is interrupted before you have a chance to get the person's name. You later want to follow up on the discussion, but without a name or other identifying information, you are unable to obtain an address or phone number. Fortunately, there is a photo database of people working in your field, and you do have a clear memory of the person's face. But, with your mental image alone, how do you go about finding a particular face in this very large database?

Suppose you have witnessed a crime. You know you would recognize the face of the perpetrator if you saw it again. The police have a computerized mug-shot system with all the images on-line, but it contains tens of thousands of images. They are able to filter the mug-shots based on gender, race, and age, but this still leaves you with thousands of images to inspect. How can you efficiently search the database to determine if the criminal you seek is present in it?

These are examples of what we refer to as *the mug-shot search problem*, the problem of searching a large facial image database starting with only a mental image of the sought-after face. Since the database is large, manually inspecting every image is impractical. In fact, although the search space is finite (so theoretically one might be able to spend the time required to look through all of it), a sequential search can still fail because the user's mental image can degrade or become confused as a result of viewing a large number of

faces [7]. What is needed is a search method that minimizes the number of image inspections required to find the face (or to determine that it is unlikely to be present in the database).

Although searching a database of criminals is the classic example, with the steady increase in the availability of on-line image data, one can anticipate many other personal and commercial uses for an efficient “mug-shot search” system. For most of us, looking up personal information, such as an address or telephone number is a daily activity. While people are accustomed to requiring text information, such as a person’s name, for a successful look-up, there are some situations in which one has no better information than a mental image of a face. Technology that attempts to assist in such circumstances is already available [5][31][25], but it is limited. These emerging systems are eclectic mixtures of innovative ideas from a variety of fields. However, little has yet been done to determine their actual effectiveness at dealing with the problem they purport to solve, and much additional work remains before such systems are likely to become as ubiquitous as phone books.

1.2 Approach

Our general approach to the mug-shot search problem is to integrate a system for creating a face (i.e., a composite system) with existing face-recognition software. The research tools we built permit a user to create a facial image interactively, and, using the face recognition software, to locate automatically other similar-looking faces in a large database. The composite method allows the user to construct a new face by patching together feature components (i.e., eyes, noses, mouths, etc.) taken from other facial

FIGURE 1. The composite D was created with the cheeks, nose, and chin from A, the mouth and eyebrows from B, and the forehead and eyes from C.

images. Figure 1 shows an example of such a composite, image D, which is constructed from parts in images A, B, and C. The basic face recognition technique we use for searching the data is Eigenfaces [16][29], a method based on Principal Component Analysis (PCA), in which images are compressed from the high-dimensional space of pixel-intensity values to the lower-dimensional space of a small set of basis vectors. Similarity between images is determined by distance (e.g., Euclidean) in the space. A search order on the database can be computed in real time by sorting the images according to their distance from a "query" face. Integrating composite creation software with face-recognition software makes possible an interactive process in which interim created composites may be used to search the data, and the results of interim searches may likewise be incorporated back into the evolving composite.

Our approach to studying this integrated system is to gather and analyze user data. Such analysis is critical to understanding and is the only way to provide reliable answers to the fundamental questions that drive the design of a mug-shot search system. We use data from user studies to answer a variety of questions posed about three primary aspects of the system:

- the metric used for determining similarity between facial images
- the search strategy, and
- the interface for constructing query images

We begin evaluating, in practical terms, the effectiveness of the Eigenface metric at mimicking human perception of facial similarity. Next, since there are potentially many ways for a user to employ a mug-shot search system, we compare the effectiveness of a variety of possible search strategies. Existing systems typically give the user a great deal of freedom in selecting a search strategy, and sometimes important decisions must be made at every step in the search. Since poor decisions can be extremely costly, it is important to give both system designers and users more guidance about what types of strategies are most successful. Finally, we examine the details of the interface for constructing composites with the goal of understanding how best to enable people to create successful query images. Since search and creation may now be used in a totally integrated fashion, this provides a rich set of design choices, giving rise to new questions about what types of interface features are most effective.

1.3 Contributions of the Thesis

This thesis provides a cohesive summary of current progress on the mug-shot search problem. We present our suite of “mug-shot search” tools, which includes ideas and technologies from a range of different sources integrated in novel ways. This suite of tools represents a cross-section of many current themes and approaches to the mug-shot search problem. We present the analysis of results of a user study we conducted with these tools. The study results show that the Eigenface metric is a useful, but imperfect model of

human perception of similarity between faces. The results also show that the lack of agreement among people about what faces are most similar to a target places a significant bound on what we can expect from any metric. The choice of search strategy must take into account the performance of the metric, and, for the Eigenface metric, we show that a simple non-iterative strategy employing a single composite query image is more effective than an iterative hill-climbing strategy employing only the database images.

To our knowledge our study is the first user-based evaluation of the ability of Eigenfaces to perform face similarity retrieval (as opposed to face recognition) in the context of a mug-shot search system applied to a large (i.e., 4500 image) database. We describe several methods for analyzing and evaluating the effectiveness of a metric for mimicking human judgements of facial similarity. Applying these ideas to the Eigenface metric, we confirm that Eigenfaces provides measurable benefit in reducing the average number of image inspections required in a search (as compared to sequential search) and we quantify that benefit. Our analysis of the study data demonstrates the degree to which the performance of any similarity metric is limited by the lack of agreement among people themselves about judgements of facial similarity. Using the consensus of a group of people to define the “human” similarity metric, we compare the Eigenface metric to individual human beings at making similarity assessments that best capture the consensus of a group. This comparison provides intuition for the performance of Eigenfaces relative to the upper bound on it imposed by the lack of a single “human” similarity metric.

By means of simulations we demonstrate that, if indeed there really were a single human metric for assessing facial similarity, and if the Eigenface metric correlated perfectly with this human metric, then simple interactive hill-climbing in the space of the database images would be an excellent search strategy that could reduce the average number of image inspections required to about 2% of the database. This is fewer than for any of the other search strategies we studied. But the study data also show that this superiority of the hill-climbing strategy in principle is not sustained in practice, given the actual level of correlation between the Eigenface similarity metric and the “human” one. The study found that the average number of image inspections required in the course of the hill-climbing strategy was, in fact, closer to 35% of the database. Although this is an improvement over the 50% required, on average, for a simple sequential search of the entire database, it is not good enough for practical use.

On the other hand, given the actual performance of the Eigenface metric, we show that a non-iterative strategy of constructing a single composite query image from a set of 100 random database faces is superior to the strategy of hill-climbing with query images limited to those in the database. Study results show a simple composite strategy can reduce the average number of image inspections required to about 20% of the database, a number that begins to cross over into the realm of practicality. With these and other examples, we demonstrate the benefit of combining Eigenfaces with a composite creation system.

Finally, we present a summary of new user-interface issues that arise from the integration of composite creation and database search. We describe several particularly interesting interface ideas, illustrate them with examples, and discuss their specific advantages, disadvantages, and trade-offs.

1.4 Outline of Dissertation

Chapter 2 presents background and related work, and outlines the key research questions addressed in this thesis. Chapter 3 describes the suite of mug-shot search software tools employed in our human study, subsequent analysis, and other experiments. It also gives our general approach to evaluating and gathering user data and presents the specifics of the studies conducted. Issues and approaches pertaining to all experiments are discussed, with details of some individual experiments left for the later chapters in which they are most relevant. Chapter 4 evaluates the degree of success of the Eigenface metric in mimicking human perceptions of facial similarity, and demonstrates the potential value of combining the Eigenface metric with a system for constructing composite faces. Chapter 5 defines a set of search strategy options that system designers may support and users may employ. Via computer simulations and analysis of user-study data, we evaluate these strategies and compare their effectiveness in the context of a large mug-shot search system, and we explore the impact of the Eigenface metric on the choice of search strategy. In Chapter 6, we address query interface design issues, describing several interesting ways in which the integration of the search mechanism is connected to new interface design options. In Chapter 7, we present a summary of our conclusions and outline some outstanding research questions.

Chapter 2
Background and Related Work

2.1 Introduction

The quantity of on-line image data is constantly increasing, but finding a particular image in a large database of images remains a difficult problem. Images can be annotated with descriptive text and retrieved with traditional text-based query methods, but creating annotations requires substantial manual effort, and the annotations are rarely sufficient to capture fully the content of an image. *Content-based image retrieval* systems [13] attempt to overcome the problems of text-based searching by permitting a user to specify image attributes in ways that are more direct and natural than the English–language–like specifications required by traditional databases. One powerful approach is to let the user express the query with images rather than words (i.e., an *image-based query*). The system automatically compares the query image to those in the database and retrieves the most similar ones. This approach, which can be studied independently from and used in conjunction with text-based methods, is our general focus. While our research addresses the specific problem of content-based retrieval in facial image databases (and research indicates that human beings process faces in a specialized way [10] [9]), it is related to research on content-based retrieval in general image databases as well.

Face recognition systems typically use image-based queries to solve identification problems. The input to the system is a digital image of a face to be identified, which the system then attempts to match to images of known individuals in the database. The mug-shot search problem differs from the face recognition problem in several important ways.

One obvious difference between the two is that in the mug-shot search problem there is no digital image available at the outset to serve as a query.[1] Work in the field of computerized facial image synthesis is relevant in order to be able to *create composite faces* that may be used as queries.

Thus, our approach to the mug-shot search problem is an eclectic one, combining ideas from three different areas, *content-based image retrieval*, *face recognition*, and *composite face creation* (facial image synthesis). This chapter describes related work in each of these fields and discusses several mug-shot search systems that have recently been reported.

2.2 Eigenfaces and Other Face-Recognition Techniques

Eigenfaces [16][29] is a face-recognition technique based on principal component analysis (PCA). Using PCA, images consisting of N by N pixel intensity values are compressed from the high dimensional space of the N^2 pixel values to the much lower dimensional space of a small set of basis vectors called *eigenfaces*. The output of the PCA computation is a set of basis vectors that are ordered by the amount of variation they capture in the dataset. The basis vectors are computed from a training set rather than the full dataset, so, obviously, the size of the training set and the variation within it are factors that can affect the quality of the results. If the training data is sufficiently representative of all faces, then new faces (i.e., those not in the training set) can be represented well using the same set of basis vectors.

---

1. This and other differences and their implications are discussed fully in Chapter 4.

Each face in the database can be roughly reconstructed as a weighted sum of the eigenfaces. The weights (or coefficients) are used to determine the distance (e.g., Euclidean) between images. By projecting a facial image into the subspace of a small set of the best basis vectors, it becomes possible to compress the full information about the appearance of any face down to (ideally) the most salient information. An input face is recognized by computing its coefficients (i.e., by projecting it onto the subspace) and then comparing these coefficients to the precomputed coefficients of the database faces. If the distance between a known face and the input face is below a certain configurable threshold, the two faces are presumed to be the same person. Since some of the variation in a set of facial images may be due to pose, lighting or other characteristics of the photograph that are not relevant to the appearance of the face, it is possible that some of the basis vectors are really capturing information that is not helpful for recognition. It may also be the case that the human recognition system uses different features from those "chosen" by the PCA computation. The simple Euclidean distance metric (in which the all the vectors are equally weighted) might be improved upon by appropriately chosen weightings for the vectors. An attempt to do this is described by Wahid [30].

Prior to performing the PCA computation on a set of images, the images must be registered in some way, for example, by lining up all the eyes (i.e., the images are scaled so that all the eyes are effectively "hung" on the same two fixed-distance posts going right through the pupils). What gets lost in this type of image registration is the relative sizes of faces and features, as well as the relative distances between features (e.g., the distance between the eyes). One example of this problem occurs in a database that includes both children and adults. In such a dataset, some of the size information that would otherwise

help Eigenfaces to distinguish between the child and adult faces is lost in the image registration even before the PCA computation is performed. One possible approach to this problem is to use PCA in a hierarchical fashion by segmenting the face into eye, nose, and mouth images (for example) and performing the PCA computation on these subimages [23]. Some of the issues that arise in using this *Eigenfeature* based approach are how to select the various features to use and how to weight the corresponding eigenvectors for each feature.

Face recognition researchers have also attempted to extract automatically shape model parameters that can be used to produce outline drawings of facial features (e.g., eyes, mouths). Yuille et al. describe deformable templates [32][33] and Lanitis, Taylor, and Cootes describe point distribution models (PDMs) [19], both of which are used to perform this task. The techniques are generally slow and such line drawings have been shown to be much less effective for enabling human recognition ability than the original photographs (although research demonstrates that altering veridical line drawings to exaggerate the differences from an average face, as in a caricature, makes the drawings easier for people to recognize [28]).

Shape models (or point distribution models, also based on PCA) may also be used in combination with Eigenfaces [19]. The shape model is automatically fitted to the input face (capturing the shape and locations of the chin, nose, mouth, eyes), and the image is then warped to fit a precomputed average-shaped face to produce a *"shape-free face"*. The eigenface representation is then extracted from this shape-free face. The reconstruction proceeds similarly — the shape-free eigenface representation is

reconstructed and this image is then reverse-warped appropriately to fit it back to the shape of the input face. One advantage to combining these two representations is that extracting the eigenface representation from a shape-free face may make the technique more robust with respect to differences in facial expression and pose. Separating out the shape information from the grey-level information may also help with the problems described above in which the relative size information of features and whole faces is lost.

Graph Matching [18], a different face-recognition technique, uses Gabor wavelets and is claimed to be especially robust to differences in irrelevant features such as pose and expression [17]. One disadvantage to this method is that the "distance" computation is computationally more expensive than the simple Euclidean distance calculation that may be used with Eigenfaces. For this reason, Graph Matching may be less practical for some kinds of interactive applications.

Although face-recognition techniques were originally designed for face identification applications, they are used in mug-shot search systems for retrieving similar-looking faces (i.e., for similarity retrieval). The next section discusses systems for performing similarity retrieval in general image databases.

2.3 Content-Based Retrieval Systems

One well-known content-based retrieval system for general image databases is the QBIC (Query By Image Content) system [11]. QBIC represents images with a variety of automatically and semi-automatically computed features including average color, color distribution, mathematical representation of texture coarseness, contrast and directionality, edge maps, area, circularity, eccentricity, and major-axis direction (for shapes). QBIC's

approach to user interaction is to provide a number of query specification dialog boxes that permit the user to compose or create the image-based query. Once the query is composed, the user submits the query and the closest matching database images are displayed.

The QBIC system enables users to construct query images in a simple, direct, and natural way. For example, users can create a rough diagram of color distributions and locations and the system will sort the database images by similarity to the diagram. In one example, a simple query specification diagram of a pink circle on a green background retrieved many images of pink flowers with background greenery. Similarly, users can create shapes, texture swatches, and color histograms, and submit these as queries. QBIC's power comes from the fact that describing a shape, texture, or color by drawing or creating it is often simpler than (or at least as simple as) describing it in words, and the result is an image object that can be directly and automatically compared against the database images.

QBIC's features might facilitate a user's ability to find images of faces in a database containing varied subject matter (for example, one might search for skin colors or head shapes), but it does not have any features that specifically address the problem of searching an image database containing only faces. It is not at all clear how one would use the system to specify and search for people with particular facial characteristics.

Another general content-based retrieval system by Jacobs, Finkelstein, and Salesin uses multiresolution wavelet decompositions of the query and database images [15]. The query image creation interface is similar to QBICs in which the user can paint a rough

sketch of the query. In this system the most similar images in the database are displayed while the query is being constructed, and this set is updated continually as the query image is modified. Because the distance computation is fast, it is possible for the user to get this type of constant feedback about the effectiveness of the query. Ideally, the user will not need to tinker any longer than necessary with the query because the target should simply appear as soon as the query gets good enough. As with QBIC, it is unclear how well the distance metric would perform at assessing facial similarity, and the system does not provide an interface that would work well for creating images of faces. Creating a specific desired face from scratch with a painting tool would be a challenging and time-consuming task. For mug-shot search, one would ideally like an interface specifically designed for creating faces, perhaps by creating them directly from the database images themselves.

2.4 Composite Creation Systems

Modern police departments now use computerized systems for creating criminal composites. Compusketch, which is a computerized version of the Photofit system developed by Penry [22] [8], is one of the systems widely used for this purpose. Photofit uses interchangeable photographs of five face parts, including forehead/hair, eyes/ eyebrows, mouth, nose, and chin/cheeks, and includes accessory items such as beards and glasses. Several related non-computerized systems (MIMIC and Identikit) also let the user manipulate individual facial parts or features [8].

Photofit and similar systems require that a witness be able to recall isolated individual facial features. On the basis of psychological studies indicating that people find it difficult to apply single-feature recall, psychologists Johnston and Caldwell concluded that a

different kind of system was needed. They state that while “humans have excellent facial recognition ability,” they “have great difficulty recalling facial characteristics with sufficient detail to provide an accurate composite” [7]. To address this problem with existing systems and to attempt to take advantage of the remarkable human ability to recognize faces, they built the FacePrints system [7], which uses a genetic algorithm and allows a crime witness to create interactively a composite face without requiring isolated feature recall ability. In FacePrints, faces are represented by a set of indices into a database of six face parts (such as those used by Photofit) together with a set of position coordinates for each part. Mating and mutation operators are defined on this representation and a traditional genetic algorithm [12] is applied, except that the fitness ratings are made interactively by a human operator. The user is presented with 30 randomly generated composites, one at a time, and must rate each according to its resemblance to the criminal. Based on the ratings, a new generation of faces is produced. These faces are again subjected to rating by the user and the new ratings are used in turn to create a new generation of faces. This cycle continues until a likeness to the criminal is achieved. Johnston and Caldwell claim that this approach is more effective than systems such as Compusketch, because it uses a recognition-based strategy rather than an individual feature recall strategy and is thus better suited to the way people remember faces.

Baker and Seltzer describe a related system for generating more free form drawings of faces that uses a set of strokes (or lines) to represent a face [3]. FacePrints could also be built using a different choice of facial image representation (or genetic encoding for a face). The representation essentially determines what faces are close to one another in the

"face space" being navigated. For example, in the FacePrints representation, a single "bit" mutation could produce a face that differs from the original face only at the nose, but that new nose may be *completely* different from the original one (e.g., we might jump from a short pug nose to a long aquiline one). Another representation might enable mutations that cause a more gradual change in the appearance of individual facial features, and this might have a substantial impact on the effectiveness of the search procedure. It is possible that a different representation would offer improvements or advantages over the FacePrints representation.

Johnston and Caldwell note that the genetic code for a face produced by FacePrints *"may provide a convenient method for searching a database of known criminals to find those that most closely resemble a generated composite"* [8]. As we have already discussed, the idea of using a compact "code" or representation for a face to compare against a database of known facial images (whose codes have been pre-extracted) is used extensively in face recognition systems. The new idea alluded to here is that one might want to use the same representation for creating faces as for searching facial image databases. This would require a representation that works well for both purposes. Several obvious questions arise from this requirement. Would the facial image representation used by FacePrints work well for database search tasks? Would the representations that have been developed for face-recognition/database-search tasks work well for a FacePrints or CompuSketch style creation system? If not, it may make more sense to use different representations for search and creation. One potential problem with FacePrints'

representation, if it were to be applied to database searching, is that two perceptually similar faces may appear representationally quite dissimilar if they should happen to be composed of different parts that are each nonetheless similar in appearance.

2.5 Mug-Shot Search Systems

Now that we have discussed all the important components, we turn our attention to existing mug-shot search systems. A few systems, some research prototypes and at least one that is commercially available, have recently been described in the literature or on the web. Photobook [24], a content-based retrieval system, provides methods for searching several types of image databases containing related sets of images, including faces. In addition to faces (for which Photobook uses Eigenfaces [29]), they include databases of tools and texture images. Photobook's designers use what they refer to as "*semantics-preserving image compression*" techniques such as Eigenfaces that enable images to be annotated automatically and sorted by content. The basic idea is that image data is compressed into a comparatively small set of "perceptually significant coefficients" which can then be used to reconstruct a lossy version of the original image. What is lost is much detail of the original, but the reconstruction maintains those features that are perceptually most important. In the case of faces, this means that the reconstruction produces a face that can be easily recognized. This approach is in contrast to QBIC, where the automatically extracted image features are not generally useful for reconstruction. Photobook's focus seems to be on finding image compression techniques that have the property of being semantics-preserving and that enable quick and easy computational measures of perceptual similarity between images. To compare images, the user can select from a variety of distance metrics (Euclidean, Mahalanobis, Fourier peak, divergence,

vector space, histogram, wavelet tree, or any linear combination of these). Distance from a query image is used to specify a sort order on the database. Typically, the user selects an initial query image from a small set of images selected randomly from the database. The system sorts the database relative to the query and presents the images to the user for perusal in this sorted order. The user then makes a new selection, at which point the database is resorted relative to the new selection. This process is repeated until the user finds the sought-after image (or, failing to find it, tires and gives up). For a database of over 7,000 images, the selection-then-sort process, which is intended to be repeated until the desired image is found, takes less than one second on a Sun Sparcstation [24].

One potential problem with the Photobook interface is that the search method for which it is designed is essentially a hill-climbing approach. As such, it is prone to problems with local maxima, and the user can wind up cycling through the same set of faces without making any further progress. Further, hill-climbing may interact poorly with a metric such as Eigenfaces that is only roughly correlated with the user's perception of similarity (causing the user sometimes to mistakenly guide the search "down" the slope instead of "up"). Another drawback to the Photobook interface is that the user's query is limited to images found readily in the database. This may be especially problematic if the sought-after face is very different from the other database images. An important advantage of the Photobook interaction method is that it uses the natural human ability to recognize faces and thus, like the FacePrints system, enables specification of the query without requiring the user to articulate or even be consciously aware of what specific facial features are being sought.

In general, Photobook's emphasis is on compression (rather than query specification) techniques, while QBIC's is on extracting multiple crude image features and enabling the user to specify those features with simple constructed images. Photobook's current interface approach of scanning sequentially through sorted lists of database images looking for something similar to the image in one's head is potentially inefficient and awkward. Meanwhile QBIC's image features do not contain sufficient information to facilitate very expressive queries such as might be required for searching on individual facial features. Systems such as CompuSketch or FacePrints allow one to create faces easily, but they do not include any database search mechanism. A mug-shot search system would benefit from the QBIC notion of an image-creation interface to construct queries, with an interface that is designed specifically for faces, merged with a Photobook style database search capability.

Phantomas, a commercially available automated facial database search system developed in Germany [25], uses Graph Matching [18] as the recognition technique and claims to work well with composite sketches as well as photographs as input. However, it does not integrate the creation and search components, and the advertised search times (11 minutes for 10,000 images on a Pentium-90 PC [25]) do not yet sound practical for interactive search; also their examples are in the context of small databases (e.g., 104 images). They do refer to an upcoming pilot project with a German law enforcement institution where PHANTOMAS will be used in a forensic application on a larger-scale database (up to 40,000 images).

Recently, several prototype systems that do integrate composite face creation techniques with database search have been reported. The SpotIt system [5] uses eigenfeatures [23], applying PCA to pre-annotated facial features, such as the hair, eyes, nose, and mouth. The creation interface produces Eigenface reconstructions from the eigenfeature coefficients. The user manipulates sliders to select the desired coefficients for each feature while the system continuously responds to these selections by updating the reconstructed "composite" image. Simultaneously, the system displays the faces from the database that it deems most similar to the composite. The coefficients from an existing database face may be incorporated into the composite. This system employs the idea described by FacePrints designers in which the same representation is used for both search and creation. No user studies applied to the system are reported, so there is no information on whether the interface requiring the user to manipulate so many sliders is practical or effective. Also, since the PCA computation extracts image features that do not always correspond to those features that people understand intuitively, this may make the system less easy to use. They do describe an interesting kind of "mating" system, reminiscent of FacePrints, in which the user can interactively produce a reconstruction that interpolates coefficients between up to three "parent" images.

Another system, CAFIIR [31], uses a combination of feature-based PCA coefficients, facial landmarks, and text descriptions to construct index keys for an image. CAFIIR's composite face creation method permits the user to construct a face from a database of feature parts by blending each part onto a template facial image whose corresponding feature is appropriately warped (using the feature landmark positions) to receive it. CAFIIR permits the user to select one or more of the retrieved images to be used as

feedback to refine the search, although these appear not to be used to refine the composite directly. A side benefit to systems such as SpotIt and CAFIIR is that, in the event the database search fails (perhaps because the target face is not present), the user is left with a composite of the face that may be used to locate the person via other means.

A study by Hancock, Bruce, and Burton [14] compares the Elastic Graph Matching recognition algorithm [18] used by Phantomas to several PCA-based approaches and suggests that Elastic Graph Matching may be somewhat better at capturing human perception of similarity between faces. As far as we know, this is the only study that attempts to evaluate various face-recognition techniques for their ability at similarity retrieval rather than face-identification. It provides useful comparative results about the relative performance of these metrics, but offers no quantitative information about the extent to which any of these metrics actually reduces the effort required to search a large database.

2.6 Discussion

Photobook, SpotIt, and CAFIIR provide a wide assortment of mechanisms for addressing the "mug-shot search problem." Although the various ideas embodied in these different systems are fascinating, attempts to evaluate their usefulness applied to mug-shot search have been limited. Most of the evidence described is either anecdotal via example or involves tests performed on relatively small datasets. Although much work has been done to evaluate various face-recognition metrics for their ability to perform identification tasks, very little has been done to assess their performance at the similarity retrieval task required by mug-shot search systems. Photobook, SpotIt and CAFIIR have interfaces that

may espouse or permit a variety of different search strategies, but little effort has yet been made to try to determine what strategies are actually most effective or to understand the interaction between similarity metric and strategy. The new idea that database search and composite creation may be integrated is used to varying degrees in these systems, but the wealth of interface ideas that the integration of these two systems makes possible is yet to be fully explored. Our work focuses on several questions that remain unanswered in the current literature:

- In practical "database search" terms, how effective is the Eigenface metric at similarity retrieval (i.e., at simulating human perception of similarity between faces) and precisely what is meant by this requirement?
- Given an *ideal* computer-based similarity metric, i.e., one that truly imitates similarity criteria of a human user, how could it be used and what reduction could it achieve in the number of images the user would have to inspect before finding a target image?
- What kinds of search strategies are most effective with an actual mug-shot search system, and how does the performance of the similarity metric affect the choice of search strategy?
- How might one fully exploit the integration of systems for database search and composite creation to produce a better query interface? What are the critical components in the design of the interface and what are the advantages, disadvantages, and trade-offs that must be considered?

In the following chapters we describe various ways in which we have applied the analysis of user data toward obtaining answers to these questions.

Chapter 3
Research Tools and Methods

3.1 Introduction

This chapter describes our research tools and methods, including software tools we built or integrated into a mug-shot search system, and the design of the user study we conducted employing this software. The software was intended for research purposes only, not as a production system. Many features and components were included for experimental purposes and are not necessarily expected to be proven useful. The focus of this chapter is on those software tools that were required for the user study. Several other elements of the system that were implemented, but not directly used for the study, are introduced or elaborated upon in Chapter 6 where we discuss the query interface.

To conduct our research, we built a simple system that integrates a query image creation method specifically designed for faces with a face-recognition-based retrieval method. Our approach to composite face creation is a hybrid one, using cut-and-paste methods similar to those found in CAFIIR, combined with random composite generation similar to that found in FacePrints (although without the genetic algorithm). For image retrieval, we use the whole-image based PCA method taken directly from Photobook. (Eigenfeature-based retrieval similar to that found in SpotIt and CAFFIR was also implemented, but not used for any of the studies discussed in this thesis.) The system maintains the original functionality of Photobook, but adds to it the ability to produce composites and to sort the database by distance from them. The creation and recognition

subsystems may be used in an integrated fashion, so that interim composites can be used to search the data, and interim database search results can likewise be used to improve a developing composite.

3.2 The Data

The database we used for testing is a subset of the original Photobook face database. We eliminated many of the multiple images of individual faces, attempting to use the one image with the most neutral expression. Our final test database contains approximately 4500 images of faces of varying gender, age, and race. We use the eigenfaces and associated coefficients as originally calculated for Photobook [24]. These include 100 eigenfaces produced from a training set of 100 images selected randomly from the database. For the study, we used all 100 coefficients to calculate the Euclidean distance between images. Our system has a parameter setting that permits using any subset of these 100 coefficients. We experimented with an interface similar to SpotIt's in which the coefficients could be selectively turned on or off or weighted by manipulating sliders. However, without any user data regarding which basis vectors were most important for similarity retrieval, this interface alone did not appear to us to be sufficient to enable a wise decision to be made between the numerous possible weightings of the coefficients. Even if the decision were only whether to turn on or off each of 100 coefficients, there are 2^{100} possibilities to consider!

The database images consist of 128^2 pixel intensity values and were already eye-aligned as a preprocessing step for calculating the eigenfaces and coefficients. In addition to the known eye locations, we added annotations for the position of the eyebrows, tip of

the nose, center of the mouth, top of the forehead, and bottom of the chin. These annotations were created by hand, although this could potentially be done automatically or semi-automatically using one of several known techniques [6][29].

3.3 Composite Creation

Composites are constructed out of face parts from images in the database. The feature annotations and eye-alignment made it possible to automatically recombine face parts from several different photographs and still get (most of the time) composites in which the pieces fit together fairly well. Starting with a background image, which determines the cheeks and ears, the remaining face parts are superimposed on this background in rectangles of predefined size (see Figure 2). Feature part rectangles may be resized slightly as necessary so that the rectangles fit tightly together (e.g., so there should not be any gap between the mouth and chin rectangle or the nose and mouth rectangle). Rectangle edges are minimally blended with the background. The location annotation of a particular feature is inherited from its source image, so the process of annotating the composites is fully automated. Although we could have allowed the feature locations to move (e.g., placing the mouth lower or the eyebrows higher), as is done in FacePrints, we traded that flexibility for a simpler user interface. The results are generally good, but due to lighting, pose, and feature size variations, some problems do arise. For example, the minimal edge smoothing is not always sufficient to blend the differences when a feature from a very dark face is superimposed on a very light face. Much of the crudeness that does arise could be eliminated with more sophisticated image blending methods or preprocessing normalization methods, such as those used or proposed in SpotIt and CAFIIR.

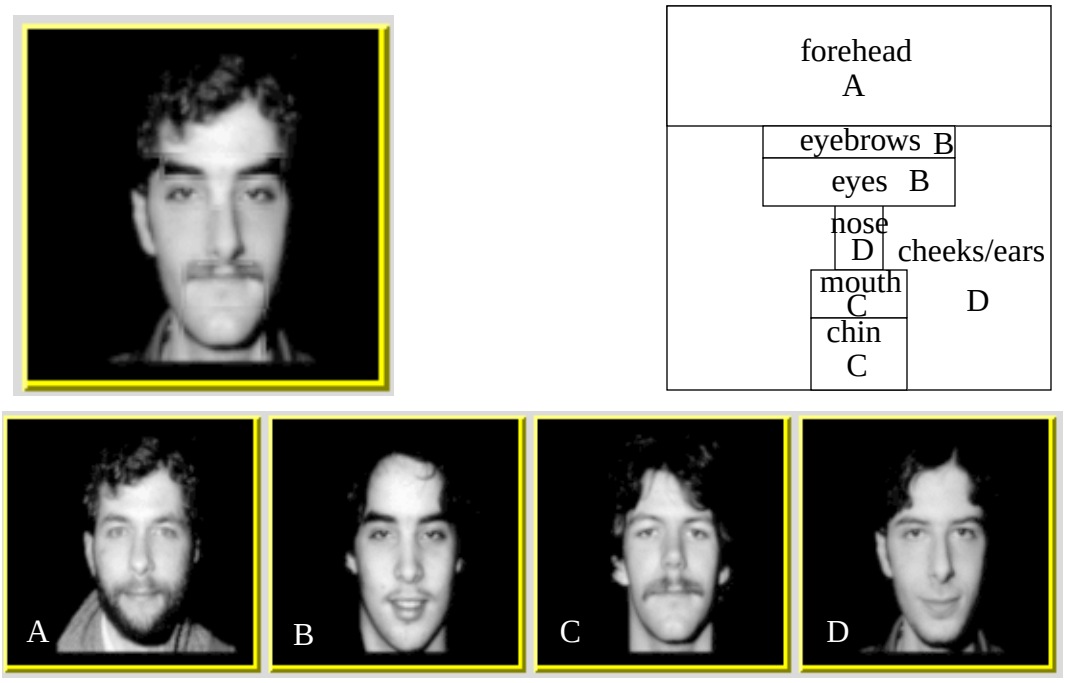

FIGURE 2. Composite Construction. The face in the upper left is a composite made up of parts from the figures below it. The forehead came from image A, the eyebrows and eyes from image B, the mouth and chin from image C and the cheeks and nose from image D. The chart in the upper right shows how the face-parts are structured into rectangles. The composite is constructed starting with the face supplying the cheeks as a background. The remaining face parts are superimposed on top of this background face, with the rectangle sizes for some parts adjusted as necessary to fit the pieces together tightly. Rectangle edges are blurred slightly with their background to smooth out edge artifacts.

The system permits composites to be constructed from parts taken from any database image or any other composite. Two basic methods for constructing a composite are available, and these may be used separately or in combination. The first method is to manually edit individual facial features. The feature editing interface we implemented requires the user to click on the desired part (e.g., nose) of an image and then click on the composite on which to place it. A drag-and-drop interface might be better, but our interface proved adequate for our purposes. An undo-edit button, which removes the most recent feature edit was added for convenience and so that users would not be reluctant to experiment with different changes. In addition to editing individual features, a composite

may be initialized with all the features from any database image. The second method of constructing a composite permits the user to work with whole faces rather than individual features. The user selects some number of faces to be used as “parent” images which the system then uses to create sets of random composites. The composites are composed of feature parts selected at random from the “parent” faces. The user may peruse the random composites looking for one that best matches his or her mental image. The random composite interface is described (along with some pictorial examples) in more detail in Chapter 6.

3.4 Why Eigenfaces?

The feature annotations made it possible to use the Photobook PCA engine to compute feature-based coefficients, such as those used in SpotIt and CAFIIR. Although we computed these feature coefficients, we elected to use a similarity metric employing only the full face coefficients for the study. There are many variations on Eigenfaces that one might use [19],[23],[30], but we did not feel that the literature contains sufficient information to enable one to make a choice between them. In the absence of any user data at the outset for comparing the many possibilities, we elected to start with full-image PCA, which is the simplest relative to other proposed approaches. The study by Hancock, Bruce, and Burton [14] indicates that using full-image PCA on a shape-free face, combined with PCA separately applied to the shape annotation data, may be better at capturing human perception of facial similarity than simple full-image PCA. Their paper was not available at the time we implemented our system. However, even if we had this information at the time, it would have been difficult to apply their shape-free technique to a database the size of ours. The shape-free approach used in their study requires 38 feature

annotations, which were created by hand for their study. We produced 6 feature annotations per image by hand, but this was a much more straightforward task. Although methods for automatically creating annotations may be possible, automatic generation of *38 accurate* annotations per image on a 4500 image database is potentially a difficult task.

We suspect that the problem with Eigenfaces (which the shape-free approach circumvents) is that although the eyes may be aligned, the scaling associated with this alignment process, as well as size characteristics of the faces themselves, can cause other features among the database images to be out of alignment. This lack of alignment means that although the PCA computation may "understand" variation in gross image features such as the shape of the face or the hair, it may be less able to "understand" the variation in some of the smaller features, such as the mouth or nose, if these features are misaligned in the data set (i.e., if all the mouths occupy very different dimensions of the pixel-space). As an alternative to the shape-free approach, we hoped that some combination of Eigenfeatures (in which the basic facial features are better aligned) together with Eigenfaces would prove sufficient to cope with this problem. Unfortunately, our informal experiments with different combinations of the Eigenfeature and Eigenface basis vectors were not sufficiently elucidating to guide a choice about how best to use these various vectors for determining similarity between images. With sufficient user data, it might be possible to analyze later whether using some subset of the feature and full-face vectors (perhaps combined with the location information contained in the annotations) would produce a better metric. Suggestions for how this analysis might be conducted from user study data are made in the future work section of Chapter 7.

3.5 Eigenfaces Applied to Composites

Since composites are produced from the original database images and inherit all their feature locations from them, the composites maintain the eye alignment and general structure of the originals. The original database images were projected onto the eigenfaces in a preprocessing step, but this operation is fast [29], and can be performed on a composite in real time. Thus, we can calculate a composite's coefficients (i.e., project it onto the eigenfaces to get its location in Eigenface space) on the fly. Once the coefficients are obtained, the database can be sorted by distance from the composite just as it can be for any database image. The entire project-and-sort operation is done in response to a single mouse-click. On a 180 MHz Pentium Pro with 64 megabytes of memory, this operation takes under a second for our 4500 image database. Note that if we had elected to use the feature vectors, it would not have been necessary to compute the feature vector coefficients for the composites, since these coefficients are already precomputed for each feature available in the database. Only the full-face coefficients need to be computed for composites. If we had elected to use the shape-free approach, the 38 feature annotations would have had to be extracted automatically from the composites in real time.

3.6 User Studies

We conducted two closely related user studies. The first was a pilot study including only 11 subjects. A second larger study (referred to here as the "final" study) included 30 subjects. Both studies were aimed at assessing the ability of the Eigenface metric to mimic human criteria of facial similarity and on understanding how the performance of the metric affects the choice of search strategy. We also strove to determine how much benefit

is obtained by adding composite creation to the system (as opposed to limiting queries to database images, as in the Photobook interface), and to gain some information about the usefulness of the interface for constructing random composites.

There are many factors that may affect the success of an interactive system (e.g., the underlying technology, the specific implementation details of the user interface and its various components, the interaction method, whether or not the system is being used by an expert operator, the strategy employed by the user, the instructions to the user, etc.). Our system has many different components, some of which may be more useful than others. If we were to conduct the test by allowing subjects unconstrained access to every component of the entire system, it would be difficult to learn anything meaningful. For example, some of the system features might be useful, but their utility could be obliterated by another less useful feature. Or perhaps the success depends a great deal on the chosen search strategy. If subjects turned out to be successful (or unsuccessful) while using the system in a completely unconstrained manner, we would have no idea why this had happened. Thus our experiments were aimed at evaluating important basic system features and strategy approaches by testing each one individually, and keeping other factors as constant as possible.

Our intended focus is on the high-level functional specification of a user-interface rather than on the specific implementation details for each function. Nonetheless, implementation details and their associated impact on ease of use can have a big effect on the success or failure of an interactive system. For example, the specific interaction method used to implement feature editing (e.g., cutting and pasting a nose from one face

to another) can have a big impact on how willing a user is to employ that function. Hoping to factor out any possible detrimental effects of our specific implementation choices, for some tasks we allowed subjects to specify their instructions to an expert operator. All subjects worked from the same automated interface that dictated the specific nature and sequence of tasks they were to perform. However, for carrying out composite feature edits and for recording their rank ordering of faces for similarity to a target, subjects could (if they preferred) specify their instructions verbally and by pointing to the screen rather than by directly manipulating the mouse themselves.

Our database could have been pre-filtered using text annotations to limit a search to images of the correct gender, age, and race. Since this type of pre-filtering advantage could be applied to all of the approaches we were comparing and would have greatly reduced our database size, we chose not to include it in our experiments.

3.6.1 Evaluation Approach

Our general approach to evaluating the system is to count image inspections. We use the *mean number of image inspections* required by the user as a scoring metric for comparisons between search strategies. We make the assumption that this metric is more important than the total time required to find a target face because a user's mental image of the target can degrade as more and more images are viewed [7].[2] We define the *score* of an image, I, (with respect to a target, T) as the position or rank of T in the list of images obtained when the database is sorted by distance from I (this corresponds to the number of image inspections required to find the target if image I is used as a query). In general,

2. While the user's mental image may also degrade over time, the time between the initial exposure to the target face and the use of the system is probably more relevant in this process than the amount of time spent at the system.

when we talk about which of several images are closest to a target in Eigenface space, we are talking not about their absolute distance from the target, but rather their image score. So when we refer to the “closest” image, we mean the one with the lowest image score.

During each study, a subject’s actions were automatically logged at every step. When a subject was asked to select database images or create composites, the scores of these images with respect to the target were automatically logged in a file. In the case of created composites, the score of each interim composite was saved as well as the bit-mapped representation of each image. This raw data provides the basis for our later analyses.

3.6.2 Target Exposure Issues

Before conducting a study of a mug-shot search system, the experimenters must decide exactly how to expose the study subjects to a target face. How does one create the mental image of a face needed at the outset? Ideally the mental image should come from a real life encounter with a person whose face is to be recalled. Although the mental image derived from a photograph may be less rich than one derived from a real life encounter, for logistical reasons we elected nonetheless to use photographs. We had access to an existing large database, but we were not in a position to locate and hire any of the people whose faces were already in it, and adding new face photographs taken under different conditions might have affected the results. Furthermore, using multiple real people for each trial would have been difficult to coordinate. Using photographs simplified our task enormously.

We also needed to decide at what point and for how long subjects should be permitted to view the target face. Allowing a subject to work directly from the target image on-screen throughout the experiment rather than from a mental image is potentially problematic because it is a less realistic simulation of the mug-shot search problem. Moreover, if one actually has an on-screen image of the face sought, the problem is indistinguishable from the face recognition problem, which is already well-studied and better solved in other ways. On the other hand, allowing the subject to view the target throughout the experiment has the advantage that it simulates a perfect photographic memory, thus creating an idealized version of the mug-shot search problem in which differences in visual memory among subjects are factored out of the experiment. This advantage may be mitigated somewhat by evidence that people's visual memory of faces plays better to holistic face recognition tasks than it does to isolated feature recall ability [7]. An on-screen image enables the subject to focus on individual features in a way that is less possible when working from visual memory. Given this dilemma, we elected to use both exposure methods. The subject performed the experiment with two different target faces. For the first target they were exposed to a photograph of a face on the computer screen and were told to study it. It was explained that they would later be asked to perform tasks that would rely on their mental image of the face. They were given several minutes to study the face. When they were satisfied with the quality of their mental image, this target face was removed from view. For the second target, on the other hand, the subjects were permitted to view the target face on screen throughout the experiment. We chose to have the subject carry out the experiment first while working from a mental image, and

Pilot Study Images

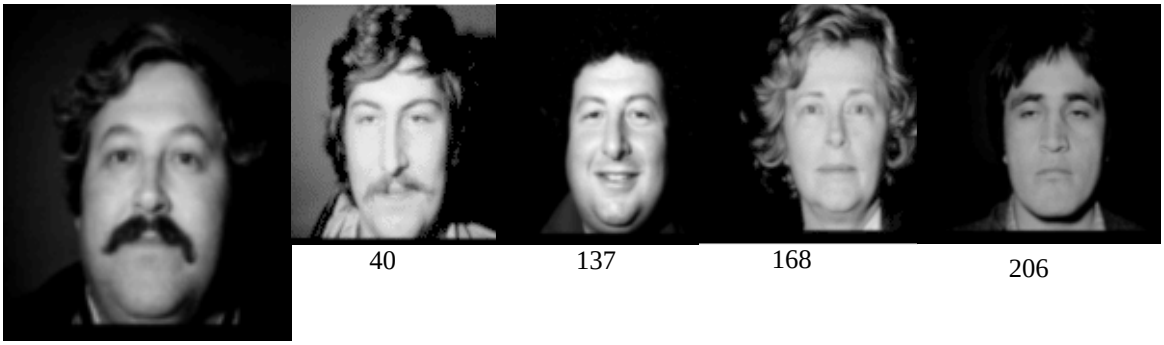

FIGURE 3. Target 1 and the four faces (out of 100 chosen randomly) closest to it in eigenspace. The number under each image indicates the number of inspections that would be required to find the target in the entire database using that image as the query.

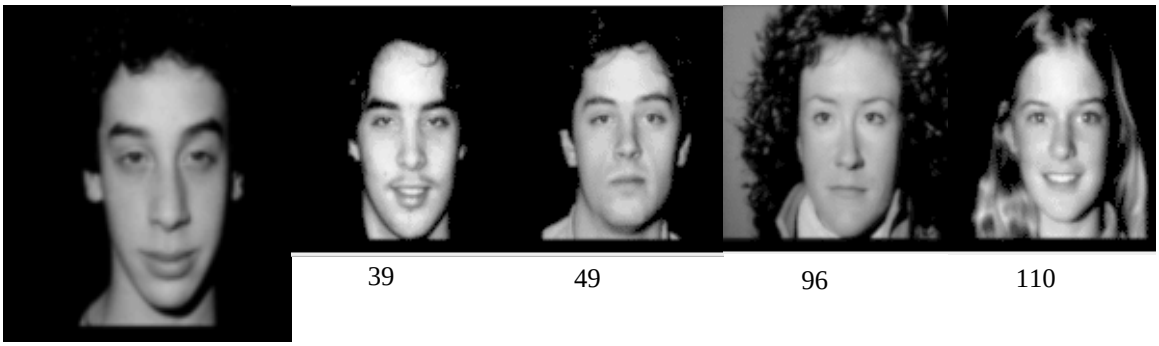

FIGURE 4. Target 2 and the four faces (out of 100 chosen randomly) closest to it in eigenspace. The number under each image indicates the number of inspections that would be required to find the target in the entire database using that image as the query.

only afterwards while working from one that was displayed continuously, on the presumption that visual fatigue might set in and cause work with the first target to degrade or confuse the ability to recall the second target.

3.6.3 Pilot Study

The pilot study included eleven subjects from our department (students and administrative staff). The same two target faces were used for each subject. Target One, shown on the left in Figure 3, was chosen specifically because the face is quite distinctive.

Target Two, shown on the left in Figure 4, was chosen at random. Also in advance, we selected 100 images at random from the database. This same random set was used in experiments for both targets and across all subjects. It was also used later for the final study and is shown in Appendix C.

After being exposed to the target face, each subject was shown the 100 random faces in a kind of computerized mug-book presentation. The screen display comfortably fits 20 faces at a time, so there were five sets through which the subject could page back and forth. The subject was asked to select the five faces from among these 100 that they thought looked most similar to the target. Selecting five was mandated even if the subject found this difficult. Once five faces had been selected, the subject was instructed to rank them for their similarity to the target, from best ("closest") to worst ("furthest"). The subject was permitted to modify the rankings (in an on-screen display of the five images in rank order) until satisfied. The four faces out of the 100 random ones that are actually closest in Eigenface space to Target 1 and Target 2 are shown at the right in Figure 3 and Figure 4. If these faces were actually selected by all users as their top four choices, one might conclude that Eigenfaces captured perfectly the human notion of similarity. (One might guess from looking at these faces that this is not the case.) Beneath each face is its image score with respect to the target. This number indicates how many image inspections would be required by a user to locate the target face if that face were submitted as a query (using our 4500 image database). We can see from these numbers that selecting the closest image in Eigenface space (out of the 100) to use as a query would enable the user to find either target in approximately 40 image inspections plus the initial 100.

After the subject ranked his or her five selections, the system generated and displayed 10 random composites created from these selections (i.e., 10 faces whose parts were selected randomly from among the subject's top five choices). The subject was instructed to select one out of these ten random composites that most resembled the target.

Finally, the subject was asked to attempt to produce a "best" composite via manual editing. The subject could start with either a database image or one of the random composites and modify its features in any way. Subjects could select facial parts from any of the original 100 faces or focus only on parts obtained from their five top choices. They could spend as little or as much time as they wanted producing a final edited composite or on any of the prior tasks. In general, subjects spent between 5 and 45 minutes on the entire set of tasks, averaging about 15 minutes per target.[3] The composite shown at right in Figure 5 is an example of a composite produced for Target 1 by a subject in the study.

FIGURE 5. Target 1 and Composite. The face on the left is target 1 and the face on the right is a composite that was created from memory of Target 1 by a study subject. Figure 1 on page 3 shows the component faces that were used to create this composite.

3. Note that, if one could inspect 100 faces a minute (and this would be pretty fast), the entire database could be searched in 45 minutes, although this process would likely be extremely tedious and error prone.

When the subject was satisfied with the edited composite for Target 1, the screen was cleared and Target 2 was displayed. The subject was asked to repeat the same set of tasks for Target 2 as those performed for Target 1. This time, however, the target image remained on the screen for the duration. Hence, for the second target, the subject worked from an on-screen image rather than a mental one.

We suspected from working with Eigenfaces that the gross image features are more important to this metric than, say, fine details of the nose or mouth. Nonetheless, when creating a composite, subjects were not given any guidance about which features were particularly important, e.g., whether to focus on gross image features or details. They were simply told to create the best composite they could.

3.6.4 Final Study

The final study was conducted with 30 subjects and 7 different targets. The targets, which are shown in the next chapter, were chosen to include both very distinctive faces and somewhat average-looking faces. The general format of the study was largely similar to the pilot study, except for one significant additional task performed for each of the two targets: After the subjects had completed all the pilot study tasks for a target, they were asked to search for the target in the database using a hill-climbing strategy similar to the one typically employed with the Photobook system. This is quite a change from the pilot study, in which subjects were never actually asked to search for the target. In the final study they were asked to do this twice, once for each of their two targets.

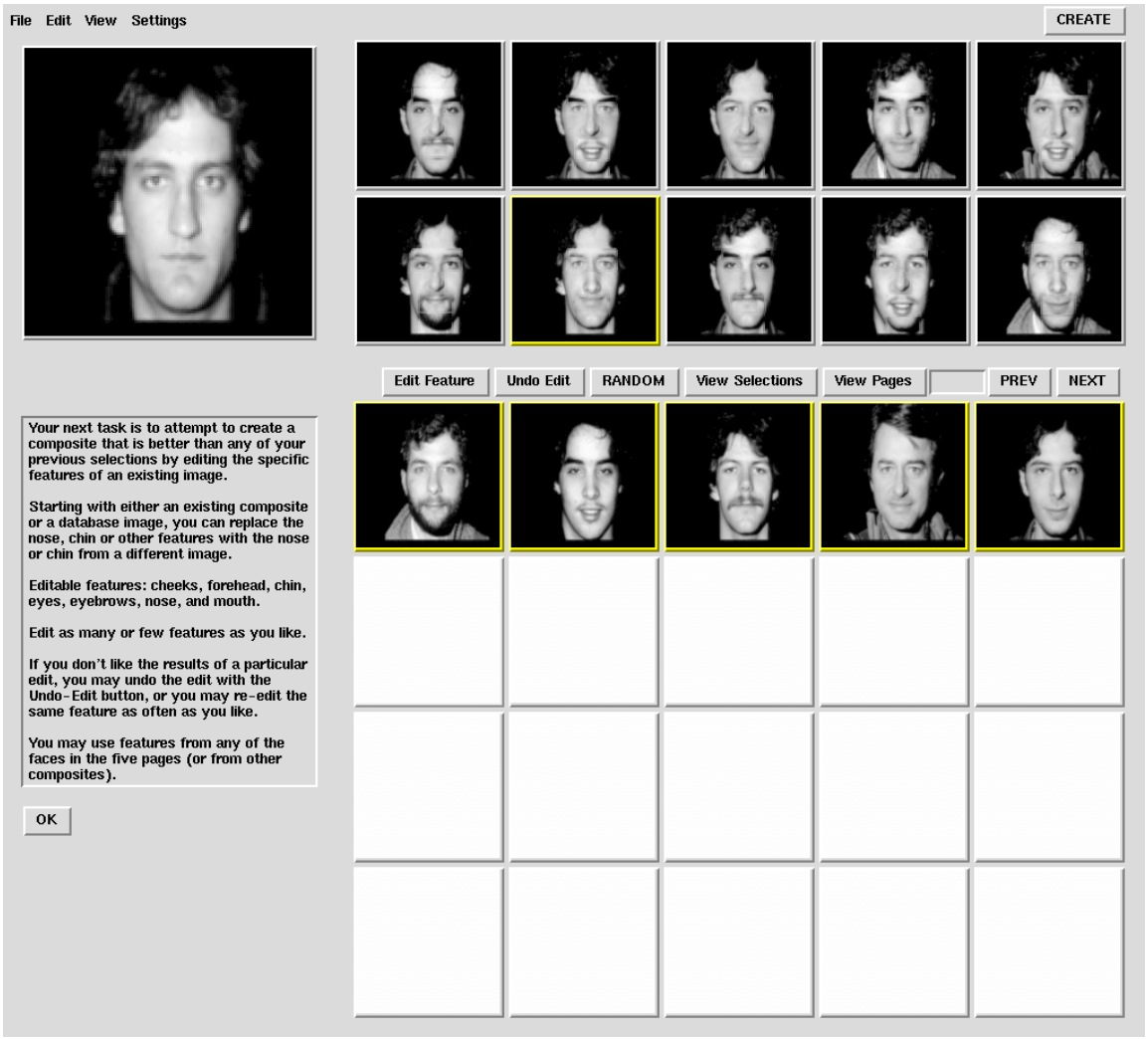

FIGURE 6. User interface for the final study. This figure shows the interface used in the final study. The target image, shown in the upper left, was removed from view for those trials in which the subject was supposed to work from a memory of it. The ten images at the top of the display are random composites constructed from the subject's five database image selections shown below them. The upper area was reserved for constructing composites via the random composite interface or via manual editing of individual features. The lower area was used for three separate views of the data: the five pages of random database selections, the subject's specific selections from among these five pages (this is the view shown), and the database itself (i.e., query results). If a view contained more than 20 faces, the Next and Prev buttons were used to page back and forth within the view. (In the case of query results, the view included all 4500 database images.) Instructions to the subject were given verbally as well as summarized in written form in the area under the target face.

The interface used for the final study was essentially the same as the interface used in the pilot study, with a few new modifications added for convenience (such as the Undo-Edit button). This interface is shown in Figure 6. Details about the layout of different areas on the screen are described in the Figure 6 caption.

After completing the edited composite, all the composites were cleared from the screen and the subject was told that the next task did not involve composites at all. Our goal for this task was that subjects should conduct a search using a hill-climbing strategy, such as might typically be applied with the Photobook interface. This task is less straightforward to explain than previous tasks, so we provide full text of the verbal instructions given to subjects:

"Your next task is to attempt to find the target face in a large database of 4500 facial images. Don't worry, you won't be required to look through the whole database! You will be restricted to a limited set of actions and I will suggest a stopping point after about 10 or 15 minutes if you don't find the face (though you may continue as long as you like, so you should not feel any time pressure). In some cases, finding the face may be a difficult task, so don't be concerned if you don't find it. We are attempting to evaluate this search method, not the people attempting to use it. As you did before, you will be selecting faces that you think look most similar to the target. This time, however, if you select a face by clicking on it with the middle mouse button, the system will respond by displaying the 20 faces in the database of 4500 faces that IT THINKS look most similar to your selected face. Don't be surprised if what the system thinks is most similar is sometimes different from what you would regard as similar. You can use the NEXT button to view the 20 next

most similar faces, and so on. The PREV button allows you to back up through this list. If you see a face that you think looks even more like the target, you can select it, and the system will sort the database again, this time relative to this new query face. If you think you are on the wrong track and want to begin again from a new set of 20 random faces, you can use the RANDOM button to view 20 randomly selected database faces. Your first action is to select a "query" face from the screen shown. After that it is up to you to decide when to use the QUERY (middle-mouse), NEXT, PREV, or RANDOM actions."

Since these instructions were long, they were followed by an opportunity for the subject to ask questions clarifying the task. In the course of this exchange (and during the search itself), we attempted to be very neutral about the depth to which each sorted list should be searched and about what criteria (other than overall similarity to the target) should be used for picking each new query image. We were not trying to be cagey. Although we developed suspicions about how best to employ this strategy, we were not certain and did not want to bias results with possibly incorrect advice. Part of the experiment was to determine whether or not feedback from the system itself was sufficient to enable people to make these decisions wisely.

For all subjects and all targets, the starting page of faces was the first 20 faces of the same 100 random faces used for previous tasks. The first query image had to be chosen from this set of 20 images. We logged the image score and the sequence of all query images selected during the course of the search. Subjects were encouraged to continue searching until they had either found the face or had inspected a total of 1000 images without finding the face. (Coded output in a background window kept the experimenter

periodically informed about the total number of images inspected.) If the target face appeared but the subject failed to recognize it, the search was terminated at that point. Occasionally a subject appeared to become very frustrated and bored with the task after not finding the target face within several hundred image inspections. In these cases, depending on the apparent level of frustration, an earlier stopping point was suggested (usually at about 750 or 800 image inspections rather than 1000). In as many cases, determined subjects elected to continue searching well beyond 1000 inspections.

3.6.5 Statistical Analysis Issues

From these study results we wish to be able to calculate various statistics such as average scores (over all subjects and all targets) of potential query images the subject might have used. For example, we might want to compare the average score of the top choice database image (out of the 100 random images) to the average score of the edited composite. If the edited composite is better, we then want to know the statistical significance of this result. We typically have 60 data points for each task because each of 30 people performed the whole set of tasks for 2 targets (once while working from memory of a target and once while working from an on-screen photograph of a different target). For simplicity, in some cases it seemed to us clearer to treat these data points as a single statistical sample of 60 points rather than two smaller samples of 30 points each. The difficulty with doing this is that, strictly speaking, using two data points per person violates the independence rule for samples. Fortunately, results for the two trials of a single person do not appear to be highly correlated. For example, consider the score of the top choice database image for the two trials of a single person. Computed from the study results, these two scores have a correlation coefficient of .015. Likewise, consider the

score of the edited composite for the two trials of a single person. These two scores have a correlation coefficient of .178. Hence it appears that it is reasonable to assume the 60 sample points are, in fact, independent, despite having come from only 30 people. For the statistical results presented in upcoming chapters, we make this assumption and often calculate P-values and confidence levels using the single larger sample.

One might also ask whether it is reasonable to treat the 60 trials as a single set, in view of the fact that half of them were experiments in which the subject was looking at a photograph of the target throughout, and half were experiments in which the subject was working from memory of the target. For most of our statistical analyses we use paired samples, so the comparisons are made between image or strategy scores achieved during the same trial, and each trial was conducted either entirely while looking at the target or entirely while working from memory of it. Of course, it might still be the case that conclusions we draw from aggregate results are primarily a consequence of one case or the other, and it is crucial to know if that is so. Fortunately, our data indicate that this is not the situation. Looking at the target throughout the experiment consistently improves scores no matter what the task. In Chapter 5 we present a series of results that lead to a final conclusion about what type of strategy is most effective with the Eigenface metric. For this final case (as well as one other), we verify that aggregating the data does not change the conclusions. We present the results first in aggregate and then split into two half groups, one in which the trials are conducted with subjects looking at a photograph of the target, and one in which the trials are conducted with subjects working from memory of the target. In both these cases, it is clear that the aggregate and the split data support the same conclusion.

3.7 Discussion

From user data gathered in our studies we can answer some of the questions posed earlier. To determine the effectiveness of the Eigenface metric at simulating human perception of similarity between faces, we can compare subject's opinions about which faces most resemble a target to the "opinion" of the Eigenface metric. We can also look at these opinions in terms of image scores (i.e., image inspections) and thus quantify how much assistance is actually offered by the Eigenface metric. We can compare the hill-climbing strategy to various non-iterative search strategies the subject might have employed using his or her top choices from the 100 random faces and/or composites as queries. By examining average search scores over all targets and all subjects, we can draw some statistical conclusions about whether or not the composites are useful and about which strategies are most successful. In the next chapter we begin by evaluating the ability of the Eigenface metric to mimic human perception of facial similarity.

Chapter 4
Eigenfaces as a Similarity Metric

4.1 Introduction

In this chapter we examine the performance of Eigenfaces at simulating human perception of facial similarity. We first present a general discussion of the important differences between face-recognition and similarity-retrieval and the need to evaluate a metric differently depending on its intended use. We describe study results indicating how successfully Eigenfaces was able to mimic "human" judgements of facial similarity, and we demonstrate the potential usefulness of Eigenfaces in the context of our system. The chapter concludes with some enlightening data on Eigenfaces' performance relative to that of individual subjects in capturing the "human" consensus when making facial similarity judgements.

4.2 Recognition vs. Similarity Retrieval

Eigenfaces was initially developed as a face-recognition method, namely, a means of identifying an input face. Face Recognition systems typically start with an on-line input image of a face and attempt to match it to images of known individuals in a database (e.g., such as in an automated entry system). This problem is distinct from the "similarity-retrieval" problem tackled by mug-shot search systems in several important ways. First a mug-shot search system does not start out with an on-line input image. In mug-shot search, query images must first be created or located. Secondly, the query image is not an actual image of the same person, but rather is a face, such as a composite, that is perceived to look similar to the target. Intuitively we understand that it is easier for people to agree

about whether two facial images are pictures of the same person than it is for them to agree about whether faces belonging to two different people look similar to one another. This intuition gives us some insight into why face similarity retrieval, though clearly related to face-recognition, is an inherently more difficult problem.

A distinguishing characteristic of the face-recognition problem is that, typically, one expects only a single correct match; hence it may not matter if the system's second-place choice looks quite different from the query image. Only the "correctness" of the first place choice is important. In contrast, a similarity retrieval system typically must retrieve a set of multiple similar-looking faces, so the "ordering" of the whole database (relative to the query) is potentially important, whereas the exact "correctness" of the first place choice is less important and less well-defined.

In addition, particular characteristics of a database can affect the outcome of face-recognition and similarity retrieval tests differently. For example, the background and lighting variations in the Photobook database could bias face-recognition results to look better, but be detrimental for testing of similarity-retrieval (i.e., bias results to look worse). This is because, in the Photobook database, typically two or more images of the *same* person have *similar* background and lighting (not to mention clothing), whereas two images of *similar-looking* people are more likely to have *different* lighting or background (or clothing), and such differences have the potential to confound the full-image Eigenface metric, making it harder for the algorithm to "see" the more important image features (i.e., the facial similarities). In contrast, when testing face-recognition, the similarities in these extraneous image features actually make it easier to match two images of the same person.

We make this point simply to underscore the difficulty of the similarity retrieval problem compared to face-recognition, and to help put some of our similarity retrieval results into a more meaningful perspective.

The significant differences between face-recognition and similarity retrieval make it clear that designers of mug-shot search systems cannot simply rely on the conclusions of face-recognition tests to guide their choice of metric. Yet, while Eigenfaces and other metrics have been studied extensively for their face-recognition capability, little research has been done to test their performance at similarity retrieval. This is not surprising, since face-recognition testing, which can be automated, is much easier to conduct. Testing for similarity retrieval, in which the standard for success is, by definition, subjective, is not possible without involving human subjects.

The results and analysis described in this thesis are one of few efforts so far to evaluate the performance of Eigenfaces at similarity retrieval. We are aware of only one prior study that addresses the topic. The study, by Hancock, Bruce, and Burton [14], provides a comparison of several metrics, including Eigenfaces, with human judgements of similarity. While their study takes on the worthy task of attempting to offer some guidance to those choosing between metrics (their results suggest that full-image PCA may not be the ideal choice), it offers little practical information regarding how the existence of a correlation between the human and system metrics actually maps into a reduction in "search score." Consistent with our results, but using a very different evaluation method, the study tells us that some degree of correlation exists. However, it does not provide a concrete sense for the amount of leverage this correlation offers. Our evaluation of the

Eigenface metric attempts to understand in practical terms (i.e., more concrete than a correlation value) how much assistance the metric actually provides users in their search efforts. Although we save the most concrete answers to this question for Chapter 5, which addresses the choice of search strategy, the remainder of this chapter presents results that begin to solidify our understanding of the value of Eigenfaces as a similarity metric.

4.3 Establishing the Foundation: Eigenfaces vs. Random Selections

In our study, subjects were presented with 100 faces chosen randomly from the database of 4500 images, and asked to select the 5 of the 100 that they perceived to be most similar to a target. *Appendix C* shows pictures of the 100 random faces used for all these experiments. One can think of the Eigenface metric as another "person" whose "choices" are those faces out of the 100 that are closest to the target in Eigenface space. We wanted to know to what extent the "choices" made by Eigenfaces agreed with the human choices when given this same task (of assessing facial similarity). Exactly how much agreement was there between people and Eigenfaces? If we want to build a mug-shot search system, how solid a foundation does the Eigenface metric provide?

To answer these questions we looked first at how often the very top Eigenface choice (i.e., the face among the 100 that was closest to the target in Eigenface space) was included among people's top five choices. For 61 trials in the final study involving 31 people (using the seven different targets employed in the final study), the top Eigenface choice showed up among people's top five choices only 21% of the time (i.e., in 13 of the 61 trials), a result that was initially disappointing. Nonetheless, this is significantly better than what one would expect if Eigenfaces had been making completely random choices.

The probability that a randomly picked Eigenface "choice" would score a hit on one of the five choices made by a particular human subject (from among the 100 faces previously selected) is precisely $5/100$. (Note that this is totally independent of which five choices the human subject made or what criteria were used in making them.) Thus people's choices included the top Eigenface choice 21% of the time, while for a random Eigenface choice, we would expect this figure to be only about 5%.

Next we asked how often people's five picks intersected with at least one of the top *two* Eigenface choices. How often did they intersect with at least one of the top *three* Eigenface choices? And so on, up to how often they intersected with at least one of the top *ten* Eigenface choices. Thus we are asking how often there is some intersection between people's five choices and the Eigenface top N choices. The upper curve in Figure 7 shows the percentage of the 61 trials in which the subject picked at least one of the top N Eigenface choices among their top five. For comparison, the lower curve shows the expected percentage if Eigenfaces were making its N choices completely at random. The error bars bracketing the subject results (upper curve) in Figure 7 show the 99% confidence interval.

The expected percentage for N random Eigenface choices (i.e., the lower curve in Figure 7) is obtained by computing 1 minus the probability that *none* of the N Eigenface picks will score a hit on one of the five human picks. The choices, of course, are made without replacement; i.e., once a face has been picked, it is not available to be picked again. For example, for N=2, we get 1 minus [the probability that the first random Eigenface choice will fail to score a hit on the human subject's five choices, times the

probability that the second random Eigenface choice will fail to score a hit on the human subject's five choices] or $1-\left(\frac{95}{100}\cdot\frac{94}{99}\right)$. The equation for general N used to compute the lower curve in Figure 7 is:

$$1-\left(\prod_{i=0}^{N-1}\frac{100-(5+i)}{100-i}\right)$$

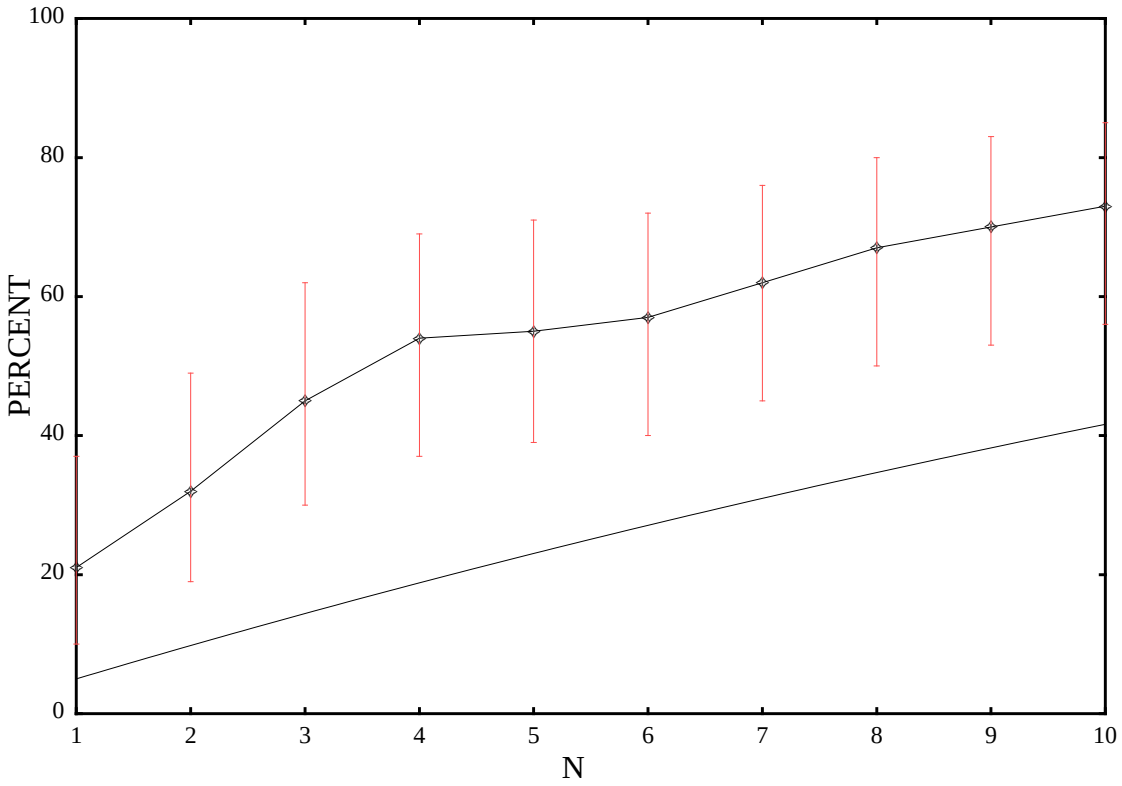

FIGURE 7. Actual Eigenface Picks vs. Random Eigenface Picks. The upper curve shows the percentage of 61 trials in which there was some intersection between the actual N closest Eigenface "choices" and subjects' top five choices (with N given on the X axis and percent on the Y axis). For example, in 33% of the trials there was some intersection between the subject's 5 choices and the Eigenface top 2 choices (as indicated by the point at x=2, y=33). The lower curve shows the expected percentage if the N Eigenface choices are picked randomly. The error bars bracketing the upper curve show the 99% confidence interval. That Eigenfaces did significantly better at intersecting with the human choices than it could have by picking its top N choices at random indicates a correlation between the Eigenface and human notions of facial similarity.

These results show a clear, but by no means perfect, correlation between the Eigenface metric and the human one. Of note is the fact that about 55% of the time, there was some intersection between the Eigenface top five choices and people's top five choices.

So far, all we have shown is that the Eigenface picks are better than random picks. The fact that there is correlation between people's choices and the Eigenface metric's is evident, but is it sufficient to be of any help in an actual search task? It appears that the Eigenface metric provides some foundation upon which one might be able to build a useful mug-shot search system. In the rest of this chapter and remaining chapters, we try to understand what type of structure this foundation best supports.

4.4 Can Composites Improve Image Scores?

The chart in Figure 7 offers some indication of the utility of Eigenfaces as a similarity metric. It provides still further help to think about these results in terms of query image scores, which give a more concrete measure of the value of Eigenfaces in a mug-shot search system. Recall that the image score is the number of image inspections that would be required to find the target if that image were used as a query on our 4500 image database. What are the image scores of the top five Eigenface choices? What are the image scores of the faces chosen most frequently by people? Suppose we ask people to construct a composite of the target out of the 100 random faces (as we did in the study), is it reasonable to expect that, at least in principle, people should be able to come up with good composites whose image scores are low? (Keep in mind that since these scores are in terms of number of image inspections, a lower score is better than a higher one.)

The next series of figures (Figure 9 on page 53 through Figure 15 on page 59) contains information that helps to answer these questions. Figure 8 provides a template indicating how to interpret these seven figures, which all have the same format. There is one figure for each of the targets used in the final study. Each one shows the target, a composite that we constructed for that target, the faces used to construct the composite, the closest faces to the target according to Eigenfaces (from among the 100 random faces), and the faces selected most frequently by subjects as looking similar to the target. The Eigenface choices are shown in order of closeness (with faces closest to the target on the left). The subjects' choices are also ordered, in this case by popularity, with the most frequently selected face on the left. Under each image is its score. (It may be helpful to convert these scores into "pages" rather than individual image inspections. For example, since 20 faces fit comfortably on a computer screen, an image score of 100 could also be thought of as the amount of effort required to look over five pages in a mug-shot book.) For the subjects' selections, in addition to image score, we state what fraction of the subjects picked the face among their top five (i.e., if 8 people worked with this target and 4 of them picked this face among their top five, the fraction is stated as 4/8). The composite *we constructed* for each target was constructed out of the 100 random faces while looking at the target face (i.e., not working from a memory of it), and with the benefit of knowing at each feature edit whether the change improved or worsened the score. We were trying not only to get a good likeness, but to do so with a minimal score. These examples are

intended simply to demonstrate that it is possible to construct a good composite out of the 100 random faces. (The score data on composites actually constructed by subjects in our study is presented in Chapter 5.)

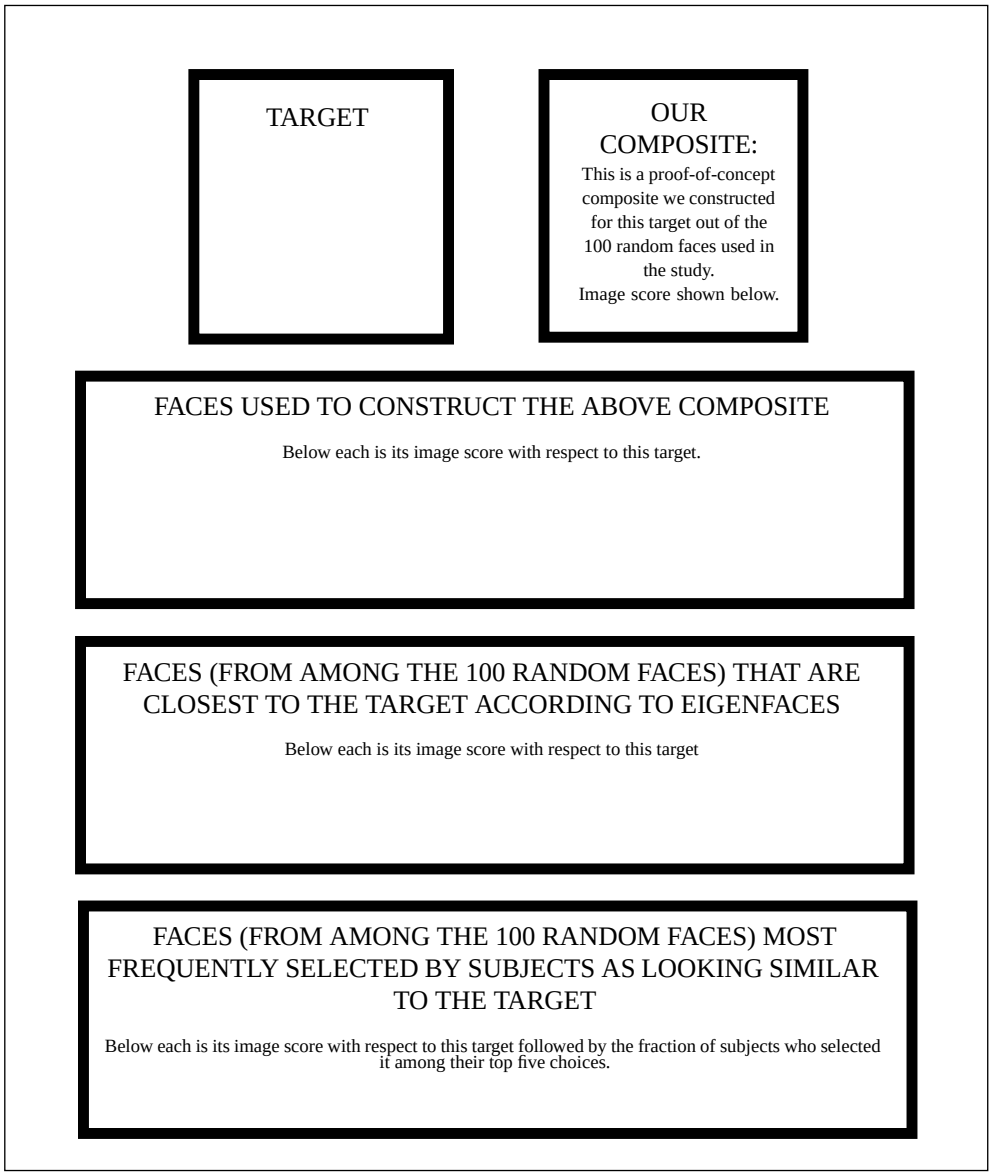

FIGURE 8. Template For Figure 9 through Figure 15. Above we give the template for interpreting the next 7 figures. There is one figure per page for each of the 7 targets used in the final study. Each figure contains the same set of information about that target as explained above.

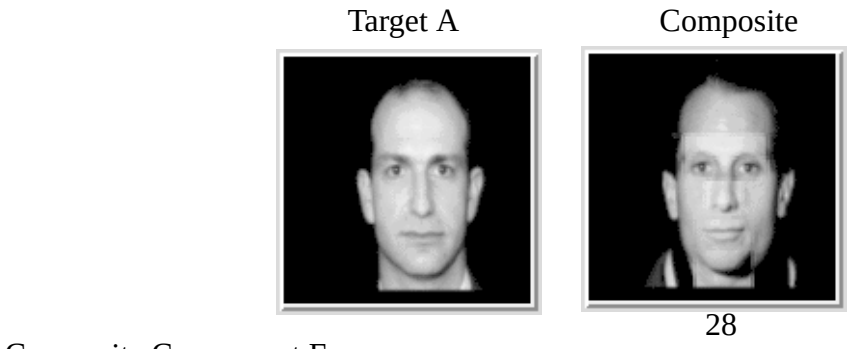

FIGURE 9. Target A (top left). It is interesting that in this case the set of faces perceived by subjects to be most similar to the target has two intersection points with the set of faces we used to construct a "close" composite. This suggests that people ought to be able to use composites successfully, perhaps employing the random composite mechanism (in which several "parent" images are used to generate sets of composites with randomly combined parts from the "parents"). It is also interesting that there is one intersection between the closest Eigenface picks and the closest subject picks, though it was chosen by only 3 of 8 subjects. Also notable is that this set of subject picks contains the only example where all 8 subjects agreed on one face (bottom left). Unfortunately, this face was not among the Eigenface top picks.

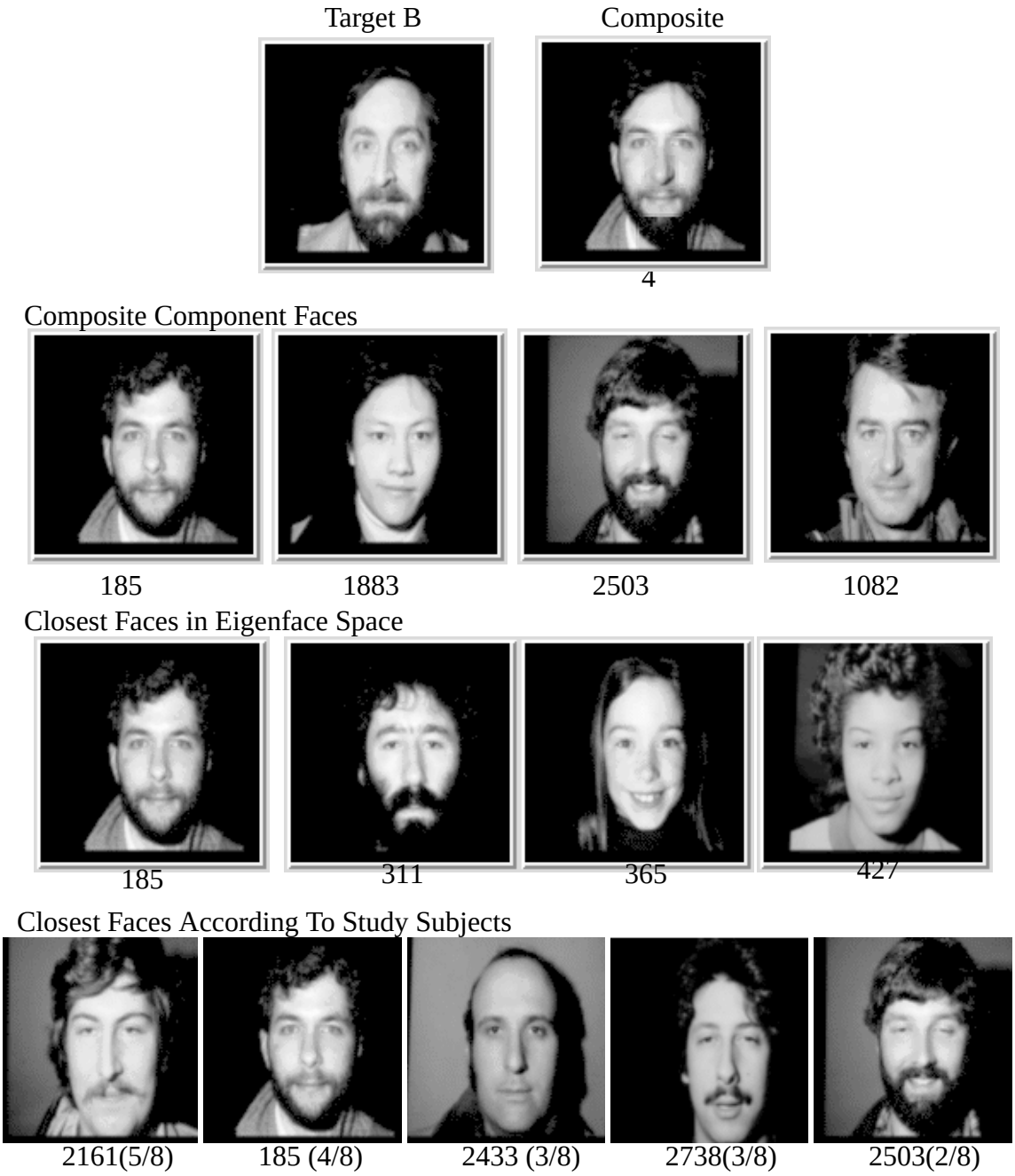

FIGURE 10. Target B (top left). In this case, half of the 8 subjects chose the Eigenface top pick among their top five, which turns out to be a fairly high level of agreement between Eigenfaces and people. The composite, which has an excellent image score (4), is constructed with a critical component, the forehead/hairline, coming from a female face (second from left in Composite Component Faces). This is an isolated feature detail that people might have difficulty recalling, but that appeared to be important to the Eigenface metric. Even when the target image was right in front of them, many subjects did not think to use a woman's forehead for a man's face.

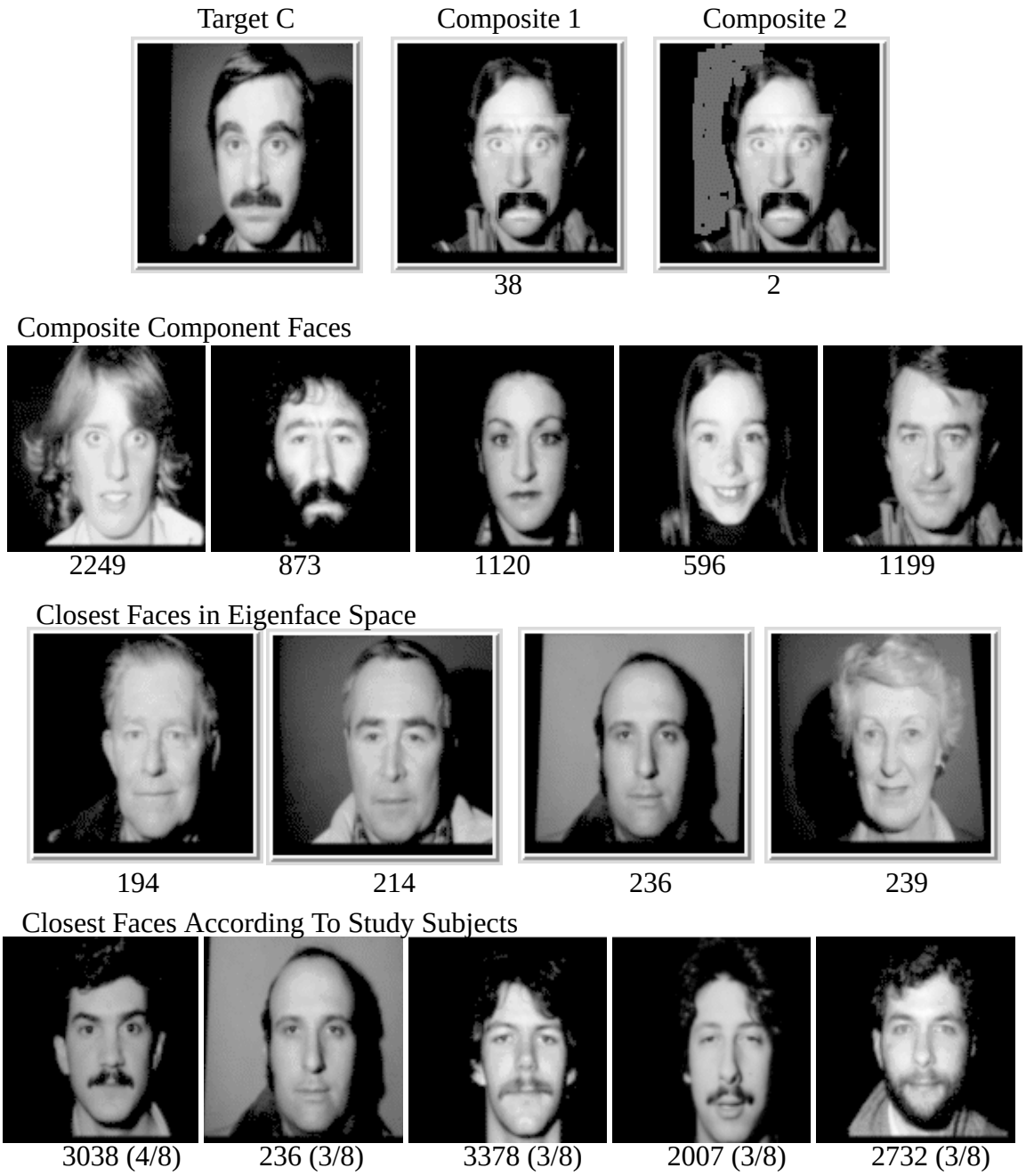

FIGURE 11. Target C (top left). In this instance, we had difficulty getting the composite image score as low as we wanted, though it was still quite close compared to any of its constituent faces (see Composite 1). We speculated that the lighter background in the target image was the trouble. We confirmed this by painting in a bit of lighter background on the composite. (Composite 2), which reduced the score from 38 to 2. Clearly, the Eigenface metric will work better if the background is masked out or if all the images have uniform backgrounds. For this target, there was one intersection between peoples' picks and Eigenfaces' and no intersection between the composite component faces and peoples' picks.

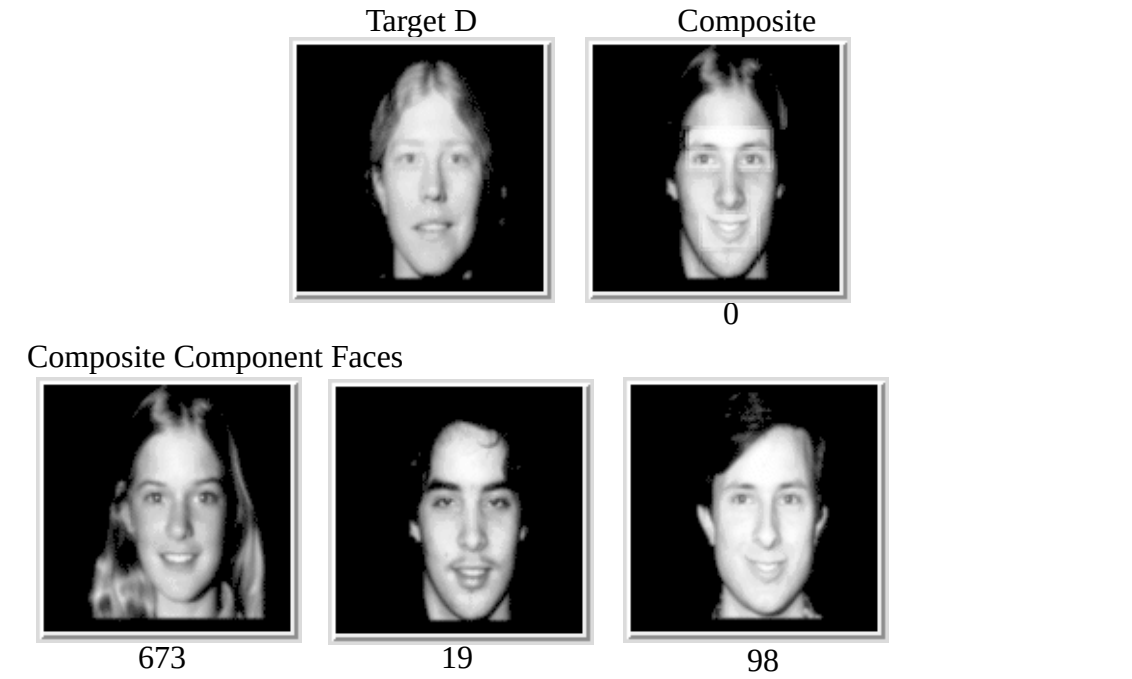

FIGURE 12. Target D (top left). Here we were able to achieve a composite with the optimal score, zero. However, it was produced with one constituent face that also had a very low score (19) and which happened to be the closest Eigenface pick as well. It appears that the shape of this face was the critical feature, causing it to be close to the target. Using this face shape as a base and adding detailed features that were fairer and more feminine seemed to be the key to getting such a close composite. For this target, there is no intersection between the people's picks and Eigenfaces. In fact, most of the Eigenface picks are male, while all of the peoples' picks are female.

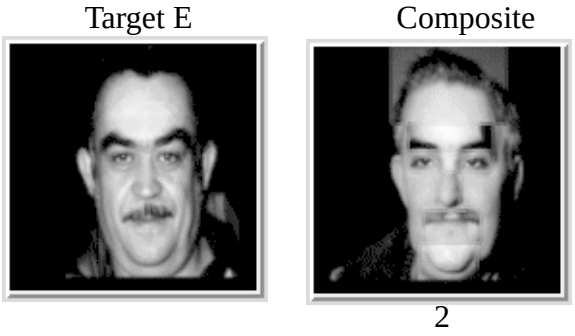

FIGURE 13. Target E (top left). The composite score is excellent in this case and is much better than any of the constituent face scores. It uses the top Eigenface pick, which has a similarly shaped face and ears to the target, but is fairer. For this target there are two points of intersection between the peoples' picks and Eigenfaces', which is quite good compared to the other targets (typically there was only one such intersection point for the other targets).

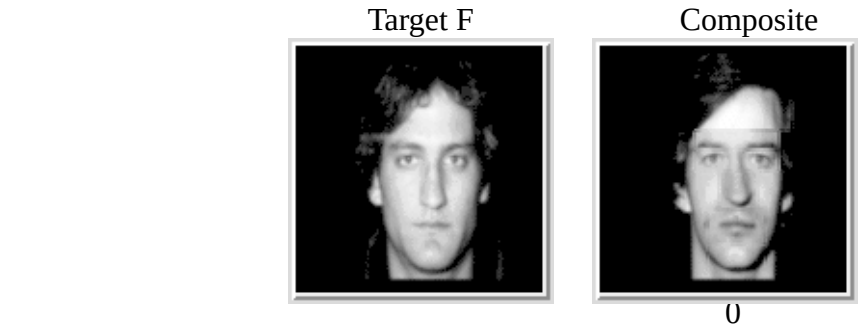

FIGURE 14. Target F (top left) The composite score is optimal (zero) in this case. The composite contains one feature (the forehead) from the third place Eigenface pick. Other than that, it was produced from constituent faces that were not among the top Eigenface picks. There is one intersection between the people's picks and Eigenfaces', but it is the least popular of the people's picks. There is one intersection between the composite constituent faces and the people's picks.

Target G

Composite

4

Composite Component Faces

56 1206 205 136 2339

Closest Faces in Eigenface Space

56 90 136 187 205

Closest Faces According To Study Subjects

56 (3/8) 302(3/8)

FIGURE 15. Target G (top left). This target face produced the lowest level of agreement among subjects, with at most three of eight people agreeing on any particular face (and this occurred only twice). The top Eigenface choice is also (one of) the top people's choice(s), but it was ranked only 3rd or 4th by the three people who picked it. Nonetheless, it provided a critical contribution (face shape) to the composite, which has an excellent score of 4.

Several things are of interest in these figures. For these seven targets, among the 100 random faces, the score of the closest image in Eigenface space averages about 100 (ranging from 16 to 195). As we shall see in Chapter 5, this is somewhat higher than we might expect for a set of 100 random picks out of a 4500 image database uniformly distributed in Eigenface-space, but does seem to be typical for the distribution of our database. This gives some sense for the type of search score one might expect from a simple strategy if the Eigenface and human metrics were identical. For example, under such circumstances, one could choose a single query image from among 100 random images and expect a total search score averaging about 200 (100 for the initial inspections and 100 for the query image).

We can also see from these figures that there is almost always (the only exception is Target D) at least one intersection between the most popular subject choices and the top Eigenface choices. Note that since we are looking at the consensus of most popular subject choices, this does not mean that *all subjects* always picked one of the Eigenface top five. We have already seen from the graph in Figure 7 that this happened only about 55% of the time.

Of particular note is that the composites we constructed all have excellent image scores and all are substantially closer to the target (in terms of image inspections) than the closest of the 100 random images. Lest we be criticized for deliberately picking target faces for which it happened to be easy to construct a composite, it is important to point out

that we did not do this experiment (constructing the composites) prior to selecting these particular targets for the study. The targets were selected without prior knowledge or any thought of this experiment.

Most of the composite scores are under 5, with the exceptions being Targets A and C, which have scores of 28 and 38 respectively. In the case of Target C, we surmised that the problem was the lighter background in the target image. We confirmed this by painting in a bit of lighter background on the composite image, which reduced the score from 38 to 2. This illustrates one of the flaws of full image PCA, although this particular problem could certainly be eliminated with a more uniform database background or by masking out the image backgrounds.

Since we were effectively "cheating" when we constructed the composites (because we had the benefit of knowing with each edit whether or not the score was improved), one might wonder about the difficulty of constructing such composites under more realistic conditions. In several cases (Target A being the most notable), there is some intersection between the set of our composite component faces and the set of most popular subject choices. This leads us to be optimistic about the potential usefulness of the random composite mechanism (the mechanism that permits the user to view sets of composites in which the features from several selected images are randomly combined). Of course, in other cases (e.g., Target C), the composites made use of components from faces that look quite different in overall appearance from the target face. The ability to synthesize features from such different-looking faces into a good composite may require an exceptional degree of visual recall and/or artistic talent.

4.5 Defining the "Human" Similarity Metric

The relative infrequency with which individual subjects included the top Eigenface "choices" among their top five choices seemed initially to suggest poor Eigenface performance at "capturing the human notion of facial similarity." However, we had not yet defined very precisely what was meant by this requirement of Eigenfaces. In point of fact, there was only limited agreement among the subjects themselves about which faces were most similar to a particular target, and this placed a limit on how well Eigenfaces or any other metric could be expected to do. It is rather more appropriate and interesting to view Eigenfaces in relationship to this limit. Looking at the actual data on how often people agreed with each other cast a different light on Eigenfaces' performance.

It was observed in the pilot study that, even when significant agreement existed among the subjects about whether to include a face in the "top five," there was little agreement with respect to ranking within the top five. Hence, we chose to disregard the ranking information. If two people included a particular face in their "top five" set, this was considered to be an agreement, regardless of how the two people chose to rank the face. With this definition of agreement, the figures on page 64 through page 67 show bar graphs that illustrate how much agreement existed among our final study subjects when they were given the task of selecting the five faces most similar to a particular target out of 100 random faces. There is one graph for each target, displaying results for the 8 or 9 subjects who used that target. The height of the leftmost bar of each graph indicates how many people selected (among their top five) the most frequently selected face. The next bar indicates how many people selected the second most frequently selected face, and so on. Thus, if there were total agreement among the subjects, the graph would look like a tall,

skinny, rectangular building, indicating that everyone had chosen the same five faces. The more squat and spread out the graph, the less agreement there was among people. These graphs illustrate well how little agreement actually existed among the subjects.

Figure 19b is a graph showing the approximate expected agreement among 5 *random* picks made by 8 people. The values in this graph were derived from a simulation in which we averaged the results from 100,000 trials using a random number generator to pick 8 sets of 5 numbers between 1 and 100. One can see from the chart that on average about 2.6 people would be in agreement about the most "popular" choice among 8 people picking randomly. This graph provides an interesting comparison against the graphs showing agreement among subjects' actual choices. Compared against the graph for Target G immediately above it, one can see that although subjects' choices for Target G are not random, neither are they terribly far from appearing random.

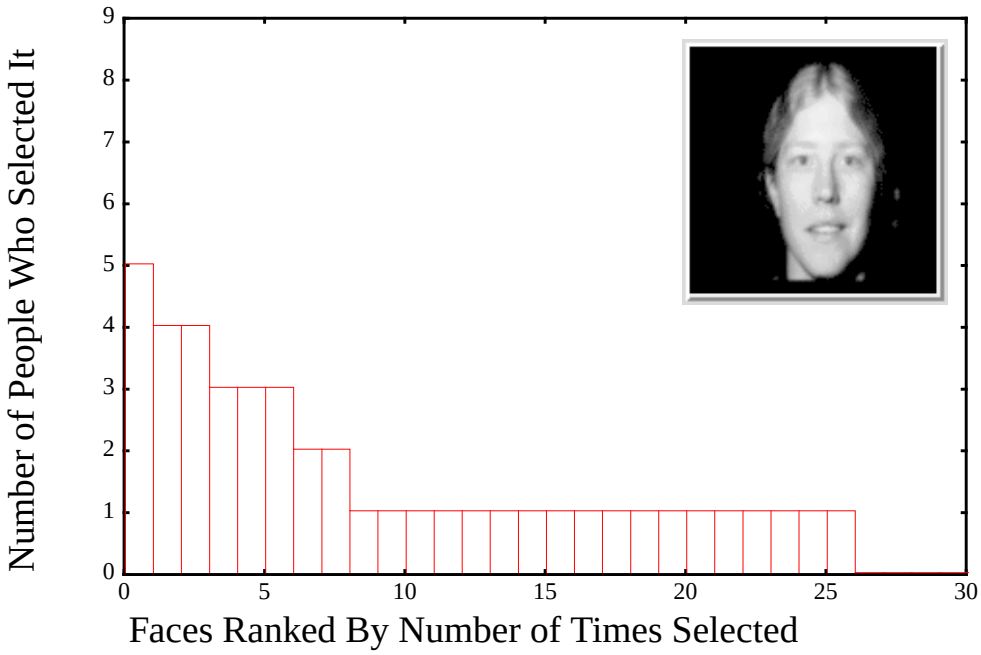

FIGURE 16a. Agreement Table for Target D (9 subjects each making 5 picks).

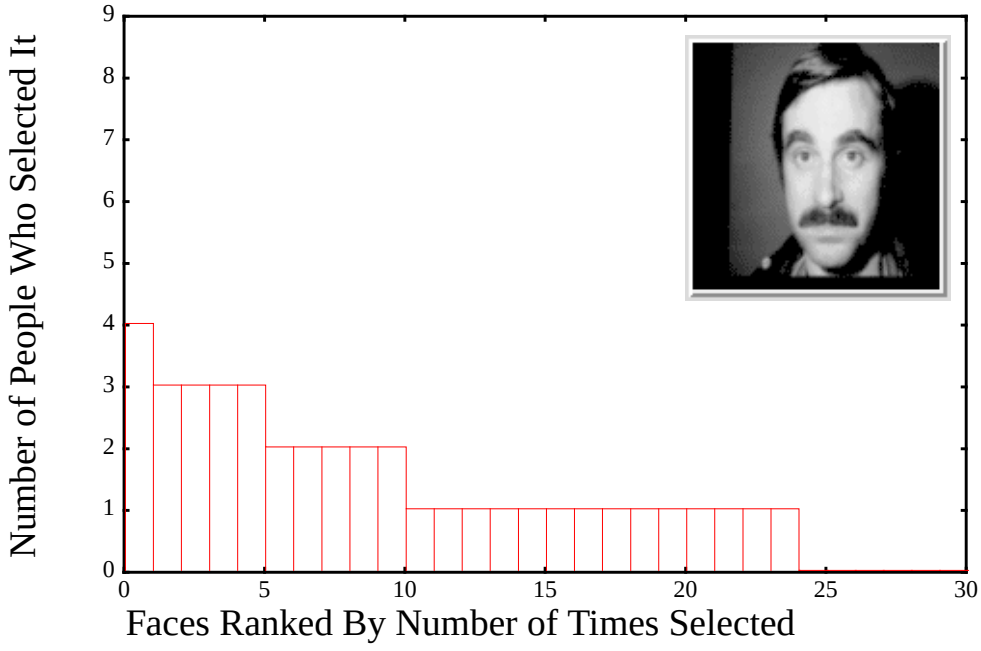

FIGURE 16b. Agreement Table for Target C (8 subjects each making 5 picks).

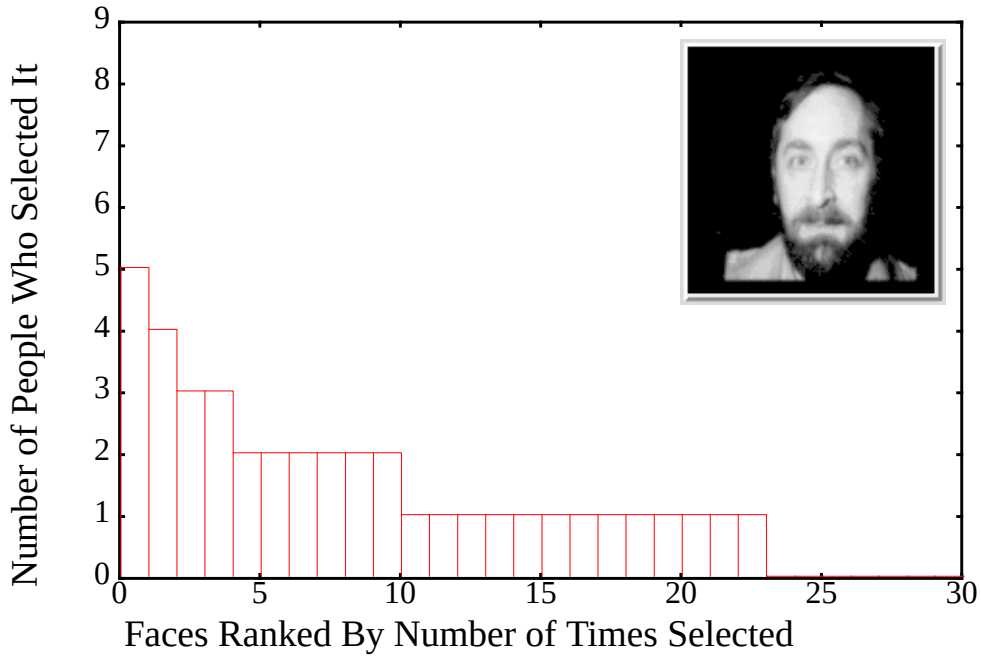

FIGURE 17a. Agreement Table for Target B (8 subjects each making 5 picks)

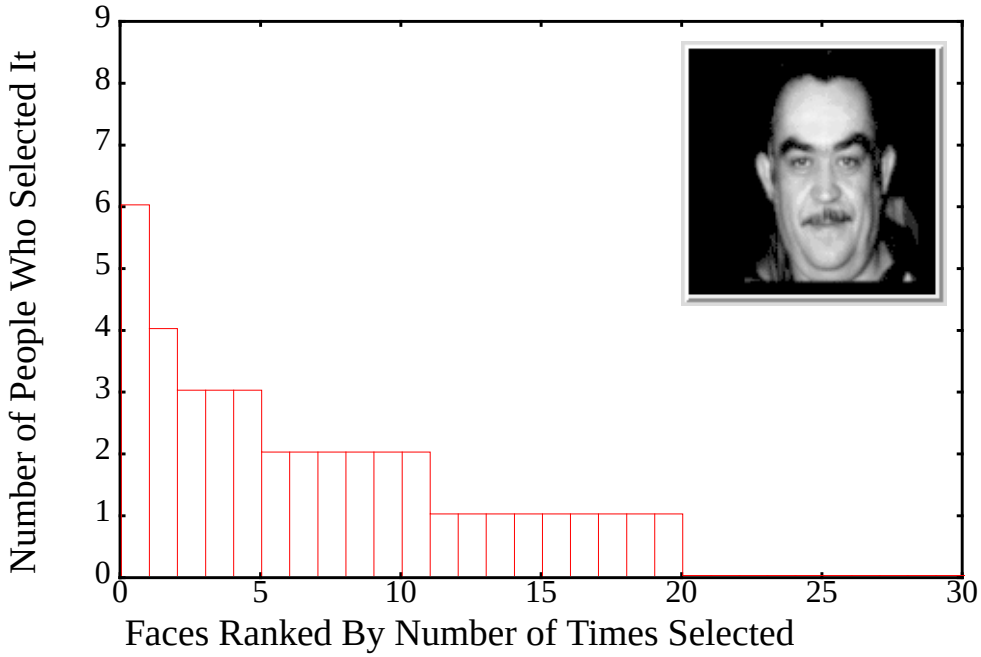

FIGURE 17b. Agreement Table for Target E (8 subjects each making 5 picks)

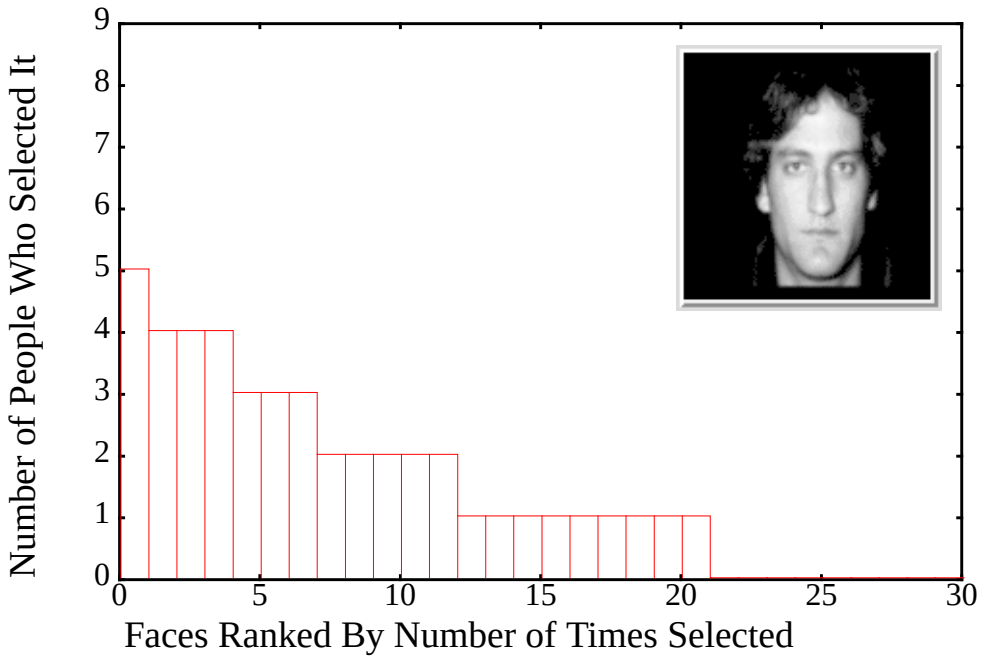

FIGURE 18a. Agreement Table for Target F (9 subjects each making 5 picks)

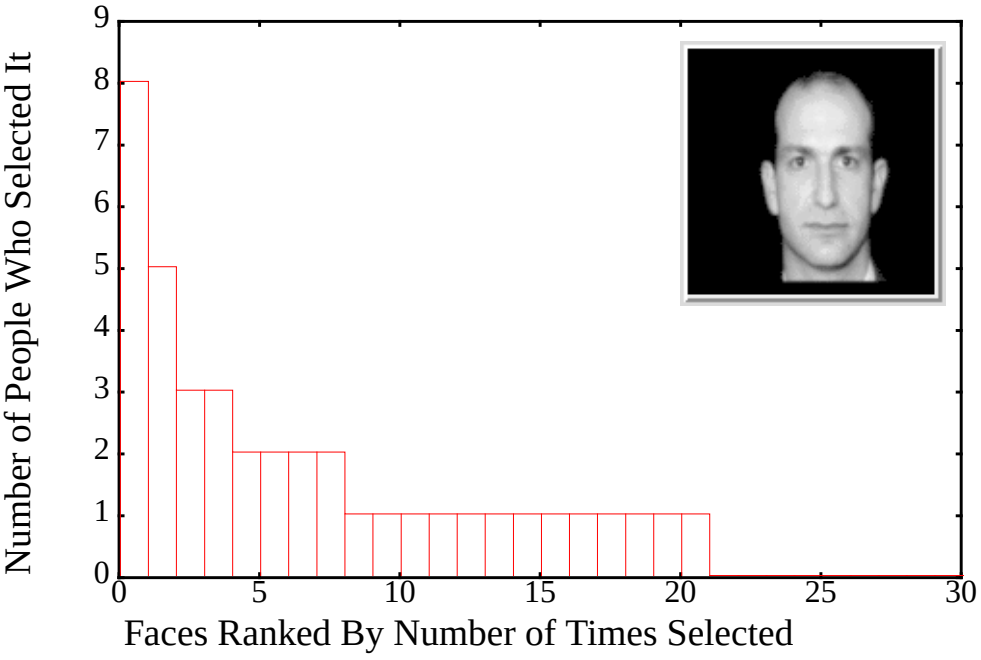

FIGURE 18b. Agreement Table for Target A (8 subjects each making 5 picks)

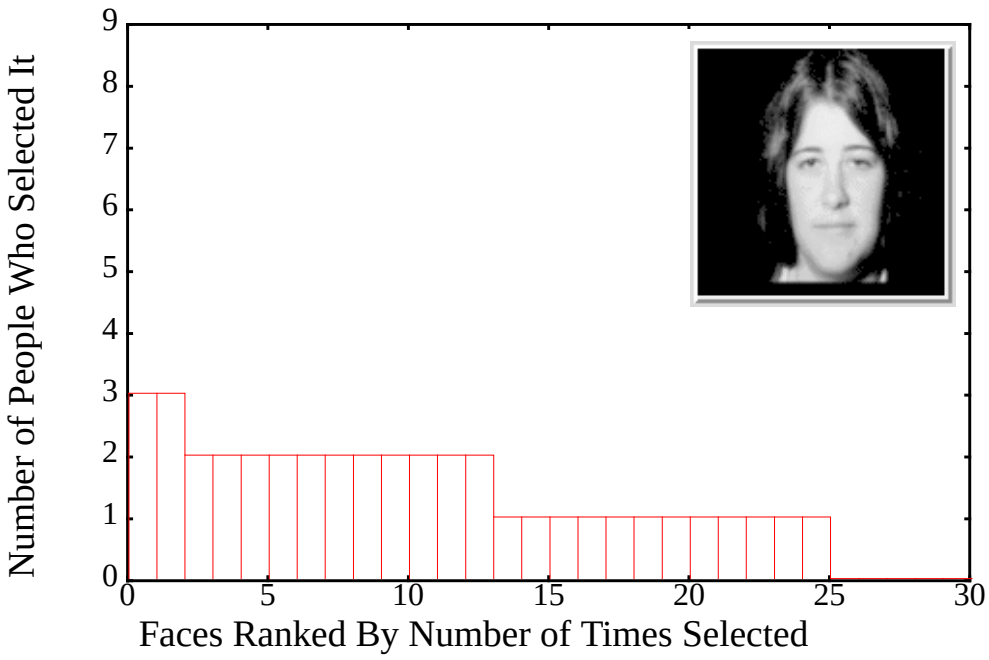

FIGURE 19a. Agreement Table for Target G (8 subjects each making 5 picks)

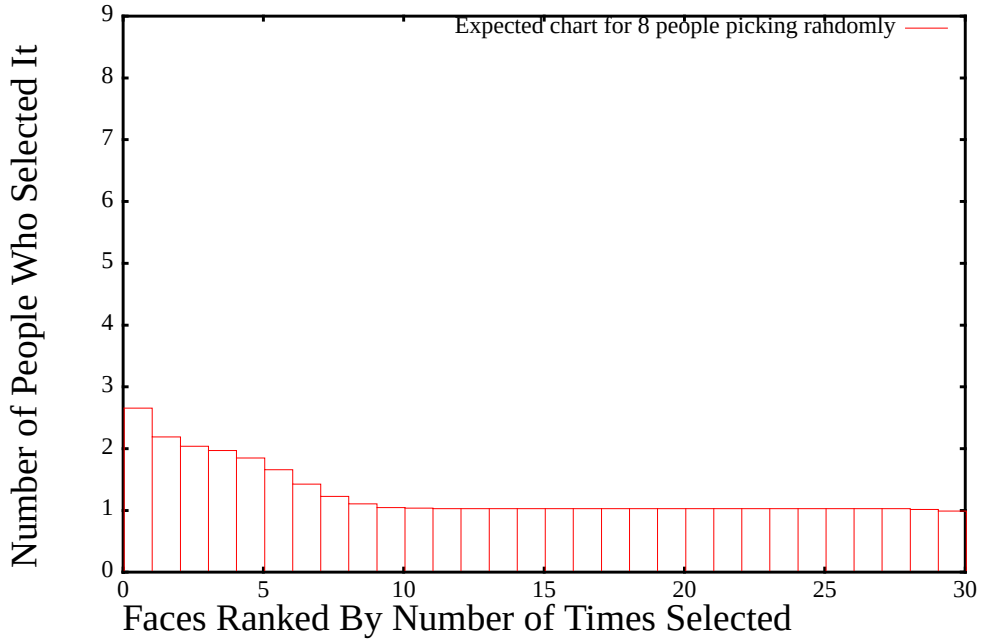

FIGURE 19b. Agreement Table for 8 People Making 5 Random Picks. This graph actually extends to an X value of 38 (tapering off to Y=0), but we cut it off so it could be more easily visually compared to the charts showing people's picks. We can see that subjects' picks for target G are closer to appearing random than for any other target.

What is being asked of a metric such as Eigenfaces is that it should be able to mimic human judgements of facial similarity. Yet this is an ill-defined requirement. What does it mean, given that there is no single universal "human" to mimic? One way of defining the requirement more precisely (in the context of this study) is to say that the Eigenface "top five picks" out of the 100 should ideally be the five faces chosen most frequently among the human subjects. However, it is interesting to note that this demands of Eigenfaces a level of performance which, out of 58 trials[4], no single subject in the study was able to achieve. In other words, no human subject in the study actually picked exactly the five faces that were picked most frequently by the group. In a larger study, this would perhaps no longer be true, but the fact that it is rare for a single human being to achieve this "maximum level of agreement" with the group is worthy of note. It suggests that we ought to have somewhat less ambitious expectations of Eigenfaces or any other metric. Although there is no reason per se to presume that a computer cannot do better at a task than a human being (there are certainly many tasks at which computers *can* do better), one does have cause to wonder whether it is realistic to expect a computer to do better than human beings at the task of mimicking human beings, and this is precisely what we are asking Eigenfaces to do.

To deal with this issue, we sought to quantify the level of "agreement with the group" of each individual subject in the study. Since we could similarly quantify Eigenfaces' level of "agreement with the group," it thus becomes possible to compare Eigenfaces'

---

4. Three fewer data points are used in the remainder of the results because three trials originally included in the study mistakenly used a parameter setting of the system in which coefficient weights were set differently from the others. In some cases it was possible to convert the data to be consistent with those of other trials, but for most of the analysis this was either not possible or it was just simpler to omit them. The three omitted data points did not appear to be unusual in any way.

performance to that of individual human beings. We define an “agreement” to be two people choosing the same face and the “agreement score” of an individual to be the percentage of the group with whom that individual had agreements. (The group is always defined as not including the person whose score is being evaluated.) Since each person in the group selects five faces, the percentage of people in the group with whom one has agreements is the number of one’s agreements divided by [the size of the group times five]. For example, in an eight-person group, if an individual (not in the group) selects one face that 6 people in the group also selected, one face that 4 people in the group also selected, and three faces that no other people selected, then, P, the percentage of people with whom that individual had agreements is $(6 + 4 + 0 + 0 + 0)/(5 \cdot 8) = 10/40 = 0.25$ or 25%.

If there is total agreement, then everyone’s agreement score is 100%. If there is no agreement, then everyone’s agreement score is 0. Since there was never total agreement in the study, an agreement score of 100% was not possible. When considering an individual’s agreement score, it is useful to compare it to the maximum possible score that could have been attained for agreement with the same group (i.e., to what we are demanding/hoping of Eigenfaces). The maximum agreement score is that of a hypothetical person who chooses the five faces selected most frequently by the group. For example, if the group is represented by the target E graph in Figure 17b on page 65, the maximum possible value for P would be $(6 + 4 + 3 + 3 + 3)/(5 \cdot 8) = 19/40 = 0.475$ or 48%. So, for this group, an individual with an agreement score of 25% has done about half as well as possible at making choices that are in accordance with the consensus of the group.

Prior to calculating an individual's agreement score, it is necessary to remove the individual from the group so as not to give credit for agreeing with oneself. Having thus defined an individual's agreement score, it now becomes possible similarly to calculate Eigenfaces' agreement score using the same group (i.e., with this individual omitted). The agreement scores of both Eigenfaces and human subjects are in this way based on agreements with the same (reduced) group, and can therefore be compared. Eigenfaces is treated as if its top five "choices" (i.e., the five faces out of the 100 that are closest in Eigenface space to the target) represent the choices of yet another individual. Table 2, which shows the results, represents a kind of sparring match between Eigenfaces and each individual subject in the study. Who can do better at making choices that capture the consensus of the group? Defined in this way, Eigenfaces did better at the task of mimicking "human" behavior than was done in 11 of the 58 human trials we conducted (i.e., in 19% of them) and as well as or better in 17 of the 58 (i.e., in 29% of them). Does this indicate that Eigenfaces has passed it's "Turing Test", appearing human-like in its behavior, if a bit on the eccentric side? Not exactly. It may be that the small size of the group is actually masking some of Eigenfaces' eccentricity, i.e., in a much larger group Eigenfaces' choices might appear more eccentric. For example, suppose Eigenfaces chooses a face that no one in the group has chosen, and suppose that a particular individual also chooses a face (a different one) that no one else has chosen. In a larger group, this individual may find others who agree with his choice, whereas perhaps Eigenfaces would

not find any cohorts because its criteria may differ from human criteria of similarity. Without a bigger sample, we don't know. Still, these results appear to suggest that the most one could ask of a better metric is somewhat less eccentricity.[5]

It is a bit disconcerting that the good Eigenface scores are limited to five of the seven targets. For the other two targets (A & D), the Eigenface scores are consistently lower than the "human" scores. For target D in particular, the Eigenface scores are especially abysmal. Target D may be the least distinctive of the seven targets, which could be a factor in Eigenfaces' performance. The problem with Target A is less mysterious. Since it provided the best rate of overall agreement among subjects (see Figure 18b), the task for Eigenfaces was inherently more difficult. The Eigenface scores for target A are fairly typical, while the human scores are unusually high.

It is also interesting to look at the mean agreement scores for both people and Eigenfaces. Table 1 shows these results, which are simply averages computed from the Table 2 data. Viewed in comparison to the mean over all subjects (as opposed to looking at contests with individuals), it becomes clear who the computer is. The mean Eigenface agreement score of 13% is conspicuously lower than the subjects' mean of 21%.

TABLE 1. Mean Agreement Scores (Final Study)

Subjects	Eigenfaces	Maximum	Random
21%	13%	43%	5%

5. It is also possible that if subjects had been told explicitly that their goal was to pick the five faces they thought others would pick most frequently, this might have produced somewhat different results and Eigenfaces might have fared less well. It would be interesting to see if specifying the task this way actually produced any better consensus among people. An advantage to defining the task this way is that it would have eliminated the problem that some participants in the study seemed to feel that the task was an "exam" at which they might be able to distinguish themselves. Paradoxically, specified this way, distinguishing oneself would be defined as not distinguishing oneself.

Reflecting the lack of agreement among people, the mean maximum possible agreement score of 43% falls well short of the theoretical maximum of 100%. Although one might wish for a metric that could achieve this 43% maximum, the 21% achieved by our subjects is the more realistic goal. While the Eigenface mean score clearly has room for improvement, there is much less room if the performance of people is regarded as the upper bound.

So that we may view Eigenfaces' performance in the context of both an upper and a lower bound, Table 1 also shows the expected agreement for five random picks (out of 100), i.e., the lower bound. The agreement score of 5% for a random picker is the same regardless of the size of the group or the level of agreement within it. If a group consists of M people (each making 5 picks out of 100), then, for any one of the random picks, the expected contribution to the total agreement score is:

$$\frac{\left(\frac{5M}{100}\right)}{5M} = 0.01$$

Since five such random picks are made, the total expected agreement score for a random picker is 5%.[6] Again (as in Figure 7 on page 49), viewing Eigenfaces' performance (13%) in relationship to a random picker (5%), we can see that Eigenfaces is doing something useful. However, if our expectation is that Eigenfaces should be able to capture the "human consensus" (i.e,. achieve the maximum possible agreement score of 43%), we cannot help but be disappointed in its performance. On the other hand, viewing the mean

6. This calculation presumes that the picking in this case is done "with replacement", i.e., that the five picks need not be unique. This is slightly inaccurate, but close enough for our purposes.

Eigenface score in relationship to the more realistic goal of average human performance (i.e., the subject mean of 21%) casts its performance in a better light.

The agreement data make it clear that the lack of a single "human" similarity metric places a confining upper bound on how much we can expect of any metric.[7] Viewing Eigenfaces in relationship to this upper bound softened somewhat our initially critical view of its performance. While there is still plenty of room for improvement, and the problem of finding a "better" metric remains an important one, the lack of agreement among people was sobering testimony to the fact that the "mug-shot search problem" is unlikely to be solved by metric alone. An equally important approach is to identify search strategies that take the best advantage of whatever limited correlation with human beings is offered by a particular metric.

7. Work by Minka [20] on searching general image databases attempts to work around this problem by allowing the user to provide feedback from which the system can infer which type of similarity metric is needed and can modify its own metric accordingly. In this way, the system is not confined to a single metric and can be more responsive to the variation among people (as well as to variation of a single person). Similarly, in the eigen-feature version of our system, the user may select any set of individual features to use as the search region, searching, for example, on just the nose and mouth region in one instance, and switching to the eyes and forehead in another. Both these approaches are attempts to work around the lack of a single human similarity metric. We have not yet conducted experiments to determine the effectiveness of the feature-based searching capability, though it has a lot of obvious intuitive appeal. It is possible that the extra complexity and added demands on the user may conspire to offset some of the benefit.

TABLE 2. Agreement Scores : Subjects vs. Eigenfaces. This table shows the agreement score results of the competition between Eigenfaces and each individual in the study. The light gray highlighting indicates cases where Eigenfaces did as well as the human subject, the dark gray indicates cases where Eigenfaces did better.

Target	Subject	Eigenfaces	Maximum Possible
A	.229	.171	.571
A	.286	.171	.543
A	.314	.114	.543
A	.343	.171	.543
A	.371	.114	.514
A	.371	.143	.543
A	.400	.171	.514
A	.429	.143	.486
F	.125	.125	.475
F	.200	.100	.450
F	.225	.100	.450
F	.225	.125	.450
F	.225	.125	.450
F	.250	.100	.450
F	.250	.100	.450
F	.250	.125	.450
F	.350	.100	.425
E	.143	.257	.514
E	.171	.286	.514
E	.229	.229	.486
E	.257	.229	.457
E	.257	.257	.486
E	.314	.257	.457
E	.343	.229	.457
E	.343	.257	.429
D	.050	.025	.475
D	.100	.000	.475
D	.125	.025	.450
D	.175	.025	.425
D	.175	.025	.425
D	.250	.025	.400
D	.250	.025	.400
D	.250	.025	.425
D	.275	.025	.400
C	.000	.086	.457
C	.086	.114	.429
C	.114	.143	.429
C	.171	.143	.400
C	.200	.114	.400
C	.229	.143	.371
C	.257	.114	.371
C	.257	.143	.343
G	.057	.143	.314
G	.086	.143	.343
G	.114	.114	.314
G	.114	.143	.343
G	.114	.143	.343
G	.143	.086	.314
G	.171	.086	.286
G	.171	.143	.314
B	.057	.200	.486
B	.143	.143	.457
B	.171	.171	.429
B	.171	.200	.457
B	.200	.171	.429
B	.257	.200	.400
B	.286	.171	.400
B	.314	.143	.400

Note: the reason Eigenfaces has 7 different scores for each target is that, for each person "challenging" Eigenfaces, we exclude them from the group before calculating their agreement score so as not to give them credit for agreeing with themselves. To be fair, we must then calculate the Eigenface agreement score with this same (reduced) group. Since each "match" is conducted with a slightly different group (with the person playing the match omitted), we get 7 slightly different scores for Eigenfaces in each case. The maximum possible score varies from "match" to "match" for this same reason.

From this table, we can see that Eigenfaces did better at the task of mimicking "human" behavior than was done in 11 of the 58 human trials we conducted (i.e., in 19% of them) and as well as or better in 17 of the 58 (i.e., in 29% of them).

4.6 Summary

In this chapter we have explained the important requirement differences between metrics for performing face-recognition and those for performing similarity retrieval. We have shown study results confirming that the Eigenface metric does indeed correlate with the “human” similarity metric, and we have acquired some sense for how this correlation might map into search scores in a large database. We have also demonstrated that, at least in principle, the method of selecting a set of random faces and using these to construct a composite, has the potential to enable users to develop a query image that is close in Eigenface space to a target. However, we have also seen that the substantial disagreement among people when making similarity judgements about faces places a significant limit on how much one can expect of Eigenfaces or any other metric. We concluded that the mug-shot search problem is unlikely to be solved by metric alone and that other components, such as the choice of search strategy, are equally important. In the following chapter, we look at different search strategies and see how the choice of strategy is affected by the performance of the similarity metric.

Chapter 5
Search Strategy

5.1 Introduction

Mug-shot search systems can permit quite a variety of search strategies. Some existing systems provide an enormous amount of flexibility, with strategy decisions that must be made by the user on the fly at every stage. For example, in addition to deciding which images to use as queries and how far down each such query list to search, the user may need to decide which, if any, of the images from these sublists should also be used as queries. The user may also need to decide whether and by what means to go to the trouble of constructing a composite. While potentially a big benefit, all this freedom can hinder the user, making the system more complicated and providing many opportunities for costly walks down blind alleys. If more thought were given to the issue of strategy up front, it might be possible to incorporate strategy decisions directly into the system, or at least to give users more guidance about which strategies tend to be most successful. In this chapter, we examine how the performance of the similarity metric affects the choice of search strategy, and we assess which strategies are most effective when used in combination with the Eigenface metric.

5.2 Strategy Definitions and Variations

One possible strategy is to select randomly a relatively small set of faces from the whole database and choose just one of these (the one that is perceived to look most similar to the target) to use as a query image. One might also consider choosing a few query images from the random set and trying each of them in turn. In principle (with an interface

that permits it), one could even search these query image lists in parallel, or search them in some order and to some depth related to the suspected likelihood of success. We refer to this whole group of strategies as the *random-set* strategy, because it involves selecting query images from a single random set and sticking with those queries only.

An alternative to the random-set strategy is to use an iterative approach, in which one tries to use each subsequent choice of query image as a stepping stone to get closer and closer to the target. For example, one might start out with a very small set of randomly chosen faces and pick the one most similar to the target to use as a query. One would then search only a short way down this list until an even better query image is found, at which point the database would be re-sorted relative to this new query. Again searching only a short way down this new list, yet another (ideally even better) query image is selected, and so on, until the target is found. For obvious reasons, we refer to this as the *hill-climbing* strategy. The Photobook interface implicitly espouses the hill-climbing strategy.

Note that both the hill-climbing and random-set strategies involve a decision about the set size. The initial set size could be predetermined in advance, or decided by the user on the fly depending on how things are going. For hill-climbing, the set size at each iteration (i.e., how far down each sorted list to look) could be predetermined in advance or could be adjusted at each step depending on the judgement of the user. This is one more example of the many decisions that are generally left to users, who may or may not have the experience or intuition to make them wisely.

One might also combine strategies, using hill-climbing for a while and then reverting to random-set if hill-climbing does not seem to be working. This might be useful if, for example, it turned out that hill-climbing worked well quickly or not at all.

Another strategy issue is whether or not to use composite faces as queries. The Photobook interface did not have a method for constructing composites, but several other recent systems [5] [31], including ours, do have this feature. The idea that composites might make better query images (as opposed to restricting users to database images for their queries) makes a lot of intuitive sense. However, as far as we know, prior to our study, no one has yet formally tested this idea or attempted to quantify the amount of benefit, if any, offered by composites. Since there is clearly some additional effort required of the user to construct a composite, it is important to know whether this extra effort is worthwhile.

Note that both the random-set and hill-climbing strategies can be used with or without composites. One might construct a composite out of the faces in the random set to use as a query. One might use one or more composites and/or one or more database images as queries. With hill-climbing, one might construct the composite iteratively, trying to refine it with features from images found in the database at each iteration.

A system that permits all of these strategies (as our research system does) gives the user an enormous amount of freedom. However, all this freedom also provides a lot of rope with which to hang oneself. Hence it is extremely important to try to determine which of these many strategies is generally the most successful. At the very least, we can offer

users some guidance about which strategies to try. And, if the results are clear enough, we might even be able to streamline the system so that it is simpler, and the user is relieved of the possibly unnecessary duty of making some of these decisions.

5.3 Image Filtering

Also at issue in the choice of strategy is whether to filter the images automatically in some way. For example, once having viewed an image and rejected it as a possible query, one might prefer never to have to look at it again. Why increase the search score by pointlessly looking at faces we've already seen? We refer to this as the *No-Review* option. Our system has a toggle that allows the user to turn the No-Review option on or off, either at the outset or in mid-search. Although No-Review seems like a good idea in principle, we have found that it does not work well in practice. There are two reasons for this. One is that it disrupts the quality of feedback from the system to the user about the effectiveness of a query. After initiating a query, the user no longer sees the faces in the database that are closest to the query image, but instead sees only the closest faces that have not already been viewed. Under such circumstances, it is harder to get a feel for the extent to which the system is "understanding" the query. The second argument against the No-Review option is that if the user once misses the target, it will never reappear on the screen. We touch on these problems again in Chapter 6 where we discuss possible interface solutions. For now, despite the fact that No-Review may greatly improve search scores, for the reasons stated above, we do not use it in our comparison of strategies.

In addition to the No-Review option, there is also the *No-Repick* option which filters out those images that have already been picked as queries. This option appears to be much more practical than No-Review, since the number of images filtered out is much smaller, and it prevents people from needlessly retrying query images. One might think that people would be able to do this filtering on their own, since it should be easy to remember which images one has already picked. We assumed that this is exactly what people would do. However, it turned out that the study subjects had significant trouble remembering which faces they had already tried as queries. Though we did not use the No-Repick option in any of the human trials, this observation caused us to conclude subsequently that No-Repick would be of real value. In the simulated trials we conducted, it turned out to be essential in order to avoid cycles.

5.4 Evaluation Method and Baseline Strategy

Recall that we define the *score* of an image, I, (with respect to a target, T) as the position or rank of T in the list of images obtained when the database is sorted by distance from I (this corresponds to the number of image inspections required to find the target if image I is used as a query). We further define the *search score* of a *strategy* as the total number of image inspections required to find the target using that strategy. Our database contains approximately 4500 images, so searching it sequentially would, on average, require a user to inspect half the database, resulting in a strategy search score of 2250. We use this as a rough baseline for comparison. Any strategy worth considering must do better than this.

We use the mean search score (i.e., the *mean number of image inspections)* required by the user as a scoring metric for comparisons between various strategies. The assumption is made that this metric is more important than the total time required because a user's mental image seems to degrade as more and more images are viewed. Of course time is not an irrelevant factor either, but the number of image inspections required clearly bears some relationship to the time required, and has the advantage that in some cases it permits one to compare several strategies without having to ask subjects actually to try them. For example, in the study, people were asked to choose database images from a random set as well as to construct composites from the set, but we did not actually make them search the lists of database images sorted by distance from any of these potential query images. This was not necessary because it was possible to calculate automatically the target's position in these lists (i.e., the image scores), and thus to compare various strategies involving one or more of these images. Of course there is always the possibility that the target would have been missed had the subject actually searched for it using a particular strategy, but this is true for all the strategies being compared, so none are unfairly penalized or favored. The use of image scores, together with the simplifying assumption that the subject would recognize the target face if it appears again, buys a lot of power in the effort to evaluate and compare differing strategies.

5.5 Best-Case Analysis

Consider the random-set strategy. For the purposes of a best-case analysis (i.e., the idealized case of a computer metric that perfectly imitates human similarity criteria), we assume that the user can immediately identify the best of the N random selections by picking the one that is perceptually most similar to the target (where "best" is defined as

the one with the lowest score). Based on a simplifying assumption,[8] it can be shown that the expected score of the best of N such random selections from a database of size (D+1) (i.e., a database consisting of elements labelled 0, 1, 2,...,D) is:

$$\frac{(D - N + 1)}{(N + 1)}$$

So, for example, given our database of size 4500, the best of 100 randomly selected images would have an expected score of $\frac{(4500 - 100 + 1)}{(100 + 1)}$, or about 44. According to this analysis, the sequential search baseline of 2250 corresponds to the expected score of a single random selection from the database, i.e., when N = 1. Clearly, the more random selections presented to the user (i.e., the bigger the value of N), the better (i.e., the lower) the expected score of the best selection. Of course, the user has to inspect all the original N randomly selected images too, and these inspections must also be included in the total search score, so here increasing N makes the search score worse, and there is a point of diminishing returns. Thus, for this approach, the optimal expected total search score is given by the minimum value of the function $\frac{(D - N + 1)}{(N + 1)} + N$, which is $2(\sqrt{D + 2} - 1)$ and occurs when $N = \sqrt{D + 2} - 1$. In our case $(D = 4500)$ the function $\frac{(4500 - N + 1)}{(N + 1)} + N$ has a minimal value when N is 66 (yielding a value of 132).[9] This means, if the user can successfully pick from among 66 random selections the one closest to the target, that pick can be used to sort the database to obtain a total expected search

8. The simplifying assumption is that the score of image P with respect to image T is equal to the score of image T with respect to image P. See Appendix B for more details.

9. Had we noted this when we originally designed the user study, we might have chosen 66 rather than 100 for the number of random images from which the user selects. Fortunately, using N = 100, we still get quite close to this minimum of 132, i.e., $((4500 - 100 + 1)/(100 + 1)) + 100$ or about 144. So our choice was also reasonable.

score of 132. This is our best case expected search score for the random-set strategy (i.e., with an ideal similarity metric), and it is quite good in comparison to our worst case baseline of 2250 for sequential search.

Doing a best-case analysis of the hill-climbing strategy is less straightforward. With the same analysis as was used for random-set we can compute the expected score of the query image at the initial iteration. After this, however, things become more complicated. Instead of attempting a mathematical analysis, in this case we resort to simulation results on our database to obtain an expected best-case search score for hill-climbing. Since we can simulate the random-set strategy as well, we can not only compare the two strategies, but also get an independent check on the accuracy of our random-set best case analysis.

5.6 Hill-Climbing vs. Random-Set

The first goal is to understand how hill-climbing compares with the random-set strategy. Is hill-climbing, as Photobook's designers seemed to assume, the superior strategy? We suspected that the answer to this question might depend on how well the similarity metric being used actually correlates with people's notion of facial similarity. However, as has already been pointed out, people are rarely in agreement about which faces among a set are most similar to a target. And even using a group's consensus, such as it is, as an ad-hoc definition of the human similarity metric, Eigenfaces still does not correlate perfectly with it. So this question has two parts. First, under conditions of perfect correlation (i.e., if the Eigenface metric correlated perfectly with the "human" metric), which is the better strategy, hill-climbing or random-set? Second, under conditions of actual correlation (i.e., given the actual level of correlation between the Eigenface and

human similarity metrics), which is the better strategy? And finally, how much benefit, if any, is obtained from the use of composites as queries? Is it worth the extra effort required to build a composite, or would users be just as well off using only database images for their queries?

We answer these questions both via simulations and via analysis of the image score results from our studies. Although the test subjects did not actually use the systems' Eigenface sorting mechanism on their composite or on their choices from the database, we apply it in a postmortem analysis of the raw user data. We sort the database by distance from each of the subject's five database selections, as well as from their first-choice composite and their final edited composite. We then note the position number of the target in each such sorted list (i.e., we note the score of each of these potential query images). (Appendix A contains these image score results.) This data makes it possible to compute average search scores across all subjects for various random-set search strategies the users might have employed. For example, we can compare how well the subjects would have done, on average, had they used only the top choice database image as a query, vs. how well they would have done had they used only the final edited composite as a query. We can also compare various "parallel" random-set strategies in which multiple images from the random set are used as queries.

5.6.1 Under Conditions of Perfect Correlation

If one assumes an idealized situation in which all people agree and the metric they are using is captured perfectly by Eigenfaces, one does not need people to perform the experiments. Instead, the strategies can be simulated on a computer using the 4500 image

face database. With the assumption of “perfect correlation,” the outcome at each step is completely determined by the algorithm and the input data (i.e., no non-determinism). For example, to simulate the random-set strategy, a target image is first picked at random out of the database, and the N random images are chosen as well. Then a check is made to see which of the N random images has the lowest image score, N_{low}, relative to the target. The final search score for this simulated data point is thus N plus N_{low}.

Hill-climbing can be simulated in a similar fashion. The simulation begins the same way, but the image with the lowest search score relative to the target (out of the initial N random choices) is used to sort the database, and the search scores of the N images at the head of this list are again evaluated to find the lowest image score (a new N_{low}). This process is repeated iteratively until N_{low} is less than or equal to N. The hill-climbing score is computed as N times the number of iterations plus the final N_{low}. By running the simulations many times with many different target images and different random sets, we can get a statistical feel for how the hill-climbing and random-set strategies compare when using the Eigenface metric (under the assumption of perfect correlation with a person).

Note that under such conditions of “perfect correlation,” the multi-query varieties of random-set (in which more than one query image is chosen from the random set) are not of interest. If we can always identify the “closest” image among a set, then there is no point in giving consideration to other possible query images in the set. By the definition of “perfect correlation” we know which query image among the N is the best, so we focus on it. Furthermore, it has already been shown that (for our database size of 4500 images) something around the square root of 4500 (i.e., 67) is a good “random-set” set size,

yielding an optimal average search score of about 132. We also pointed out that increasing the size of the set to 100 should not change the expected average score by very much. Since we ultimately want to compare with the human trials which used a set size of 100, it is preferable to use 100 as the set size in the random-set simulations as well. Before doing so, however, we confirmed that a set size of 100 is reasonably close to optimal. Mean scores for random-set simulations with a smaller set size of 67 were not significantly different from those with a set size of 100, and we also confirmed that reducing the set size to as low as 40 caused the mean scores to worsen.

For hill-climbing we also tested several different set sizes in order to determine the optimal. Tests with set sizes of 20, 40, and 60 images showed a set size of 40 to be better for hill-climbing than either 20 or 60, so we used 40 for the final simulations. Increments of 20 seemed reasonable because this represents a page or screenful of images. Scores were about 60% worse with a set size of 20 and 40% worse with a set size of 60.

Initial hill-climbing simulations revealed that cycles were common. To prevent them, we used the No-Repick option, i.e., we modified the algorithm never to reconsider a query image that had been selected previously. Once marked as chosen, a query image was omitted from any subsequent sortings, never to be "viewed" again. Without No-Repick, cycling was a frequent problem, and the hill-climbing simulations would often get stuck in infinite loops on local maxima.

Even when using the No-Repick option to prevent cycles, in some of the hill-climbing trials the algorithm still appeared to get stuck on local maxima, sometimes repeating the cycle of getting close and then moving farther away again, and other times simply making

no real progress whatsoever. We checked to see if there was anything unusual about the randomly selected target images in these cases. About a third of these failures did turn out to be cases in which the target image was anomalous, such as someone wearing a big cowboy hat or strange glasses. For the remainder of the cases there was no obvious explanation for the phenomenon. To cope with it we again adjusted the algorithm slightly in a way that could be easily adopted by a human being. In the modified algorithm, at the point when the strategy score reached 1000, we simply stopped iterating (this corresponds to a human being who, after hill-climbing to the point where he or she has inspected 1000 images, does not now pick a new query image, but instead continues searching the current query list as far as necessary, i.e., until the target is found). In such cases, we compute the strategy score by adding the score of the final query image to the cumulative search score (preceding the iteration in which the final query was selected). Since the typical hill-climbing score turned out to be close to 100, by letting the score go to 1000 before reverting to a non-iterative approach, we were satisfied that this modified algorithm gave pure hill-climbing ample opportunity to succeed before abandoning it (in fact, as is discussed later, we could probably have done better, on average, by abandoning it even sooner, but we wished to use as pure a form of hill-climbing as possible at this point). It is worth noting that the problems with local maxima occurred in about 6% of hill-climbing trials with a set size of 20, but occurred in less than 1% of trials when a set size of 40 was used. If we consider only those trials that succeeded in under 1000 image inspections, the mean scores for set sizes of 20 and 40 were essentially the same. Thus it seems the improvement in score when using a set size of 40 was primarily due to the reduction in these incidents of local maxima.

Since a tighter confidence interval is obtained for statistics involving paired samples, we set up the final simulations in pair-wise fashion, comparing hill-climbing and random-set using the same target and, to the extent possible, the same initial random set. For each data-point-pair a random target was picked as well as a set of 100 random images from the database. To keep the pairs as alike as possible (except for strategy), we used the first 40 of the same 100 random images used for calculating the random-set score as the starting set for hill-climbing.

The final strategy score results, using a set size of 100 for random-set and 40 for hill-climbing were 190 vs. 100 mean image inspections, respectively. Testing at the 1% error level, the difference of 90 between these two means is correct plus or minus 33 and is statistically significant, with $P << .0005$. The random-set score of 190 is higher than the mean of 132 predicted by our earlier best-case analysis. This is probably due to some lack of uniformity in the distribution of the database faces in Eigenface space. These results support the theory that hill-climbing is superior to random-set, at least under such conditions of "perfect correlation."

We also tried simulating a modified hill-climbing algorithm using the No-Review algorithm, meaning that any face, once viewed, was never "displayed" again. This eliminated not only those faces that had already been used as queries, but also any face that had been inspected before at all. In these simulations the No-Review algorithm did appear to produce better scores overall and to be even less prone to problems with local maxima. One explanation for this is that by refusing to reconsider a face once it had been rejected in preference to another face, we reduce the likelihood of unproductive backward

steps in the climb. A productive backward step is one that gets you off a local maximum. Furthermore, the method forces a broader sampling of the database in considering possible queries. Hence, the backward steps, when taken, are more likely to be testing an entirely new region. Despite the benefits of the No-Review algorithm, because of the real-life impracticalities discussed earlier, we chose, in our final simulations, to use the more restrictive version of hill-climbing, i.e., without the No-Review algorithm. Had we been using the No-Review hill-climbing algorithm, the superiority of hill-climbing would have been even more impressive.

To summarize our conclusions, if the "human" metric for determining similarity between images correlated perfectly with the Eigenface metric, then hill-climbing would be an excellent choice of strategy and one could expect a typical strategy score of about 100 image inspections (i.e., 2% of the size of the database). Unfortunately, we have seen that the "human" similarity metric does *not* correlate perfectly with the Eigenface metric. The obvious next question to ask is whether, under more realistic conditions, hill-climbing is still superior. How do the scores for the two algorithms differ under conditions of "actual correlation" between the Eigenface and the "human" metric?

5.6.2 Under Conditions of Actual Correlation

Using the data from our study, we can now compare hill-climbing to random-set as employed by real people. Of course there are some important differences between the simulated trials and the human trials. The random-set strategy was fairly simple to

implement, so the human data is reasonably straightforward. However, the hill-climbing algorithm is not only more confusing to explain to people, but it was applied in a less strict fashion than in the simulations.

First of all, people were told to search each query list until they found another query image that they would like to try, and that it was up to them to decide how many query images to try and how far down each query list to search before picking a new query image. Thus people were not restricted to sets of 40 images as in the simulation. Rather, subjects were allowed to choose their set size at each iteration. We hoped that by allowing people to make their own decisions, they would be able to make wise adjustments to account for good or bad luck. We had a good deal of informal evidence to suggest that when people used a restrictive set size, they would frequently get stuck in an unproductive region of the database. Although it would be interesting to run a similar study in which people were restricted to the "optimal" set size, we guessed that people would do better without this restriction. Certainly no such restriction is typically imposed in other mug-shot search systems, so it was more realistic to run the study without it.

Another difference between the simulations and the human "hill-climbing" trials is that people were allowed to back up one level if they felt they had made a poor choice of query (i.e., go back to the previous query). They were also allowed to revert to a page full of random images (i.e., 20) at any point if they felt they were going down a dead end path.

Because there were decisions to make at every step, there was some confusion among subjects about exactly what they ought to be doing. In the face of questions, we simply reiterated the options (query, random, or search) and explained that it was up to them to

decide how to apply them. We believe that any confusion we witnessed is typical of the confusion people attempting to employ this algorithm would face under real conditions as well.

Subjects were encouraged to keep trying until their search score reached 1000 before giving up on hill-climbing. In about 11 of the 58 trials people seemed so fed up with what appeared to be a completely unproductive process that we didn't feel it was fair to force them to continue beyond about 750 or 850 image inspections. In at least as many other cases, people were so persistent in their efforts that they went well beyond 1000 image inspections before giving up. For consistency, we scored failure cases (i.e., all cases in which the subject did not find the target) as 1000 plus the score of the last query image.[10]

For scoring people at the random-set strategy, we simply added 100 (for the inspections of the 100 random images) to the image score of their first choice database image (recall that subjects were asked to rank their five selections out of the 100 random database images for similarity to the target). Since subjects were not actually asked to search the query list associated with this image, there was a potential unfairness in comparing these scores to the hill-climbing scores, in which people were actually conducting a search. If, while hill-climbing, a subject passed the target without noticing it, this should not be counted as a failure, since we had no way of knowing how many corresponding misses of this sort might occur for random-set. During the study, we always

10. Rather than using the score of the very last query image chosen, we could have added 1000 to the score of the last query image chosen when the search score reached 1000. However, this method of scoring would have resulted in even worse scores for hill-climbing and would not have changed our conclusions in any way.

stopped people if they “found” the target while hill-climbing, regardless of whether or not they actually noticed that they had found it. There were several such cases, particularly occurring after people had already inspected hundreds of images.

With all these issues and differences in mind, we now turn to the results. Figure 20 is a graph of both sets of scores in sorted order (from best to worst along the x-axis, with search score on the y-axis). Random-set now did better overall, with an average score of 1431 vs. hill-climbing’s average of 1586. However, a paired t-test showed the difference of 155 to be unconvincing. Furthermore, hill-climbing was better at the low end, which is the more important end. It is also noteworthy that the failure rate (where failure is defined as a score of 1000 or worse) was essentially the same for both random-set and hill-climbing, about 59%. For our purposes the scores are not significantly different, and both are very poor. When compared to the idealized perfect metric mean simulation scores of 100 for hill-climbing and 190 for random-set (i.e., 2% and 4% of the database size), it is disappointing to see the magnitude of the negative impact of the less-than-perfect Eigenface similarity metric. Not only is hill-climbing no longer better, but both strategies, despite yielding significantly better average scores than the 2250 for sequential search, are overall so bad as to be impractical in any realistic situation. Sequential search yields an average search score of 50% of the database, while the random-set and hill-climbing results improve that to only 32% and 35% respectively.

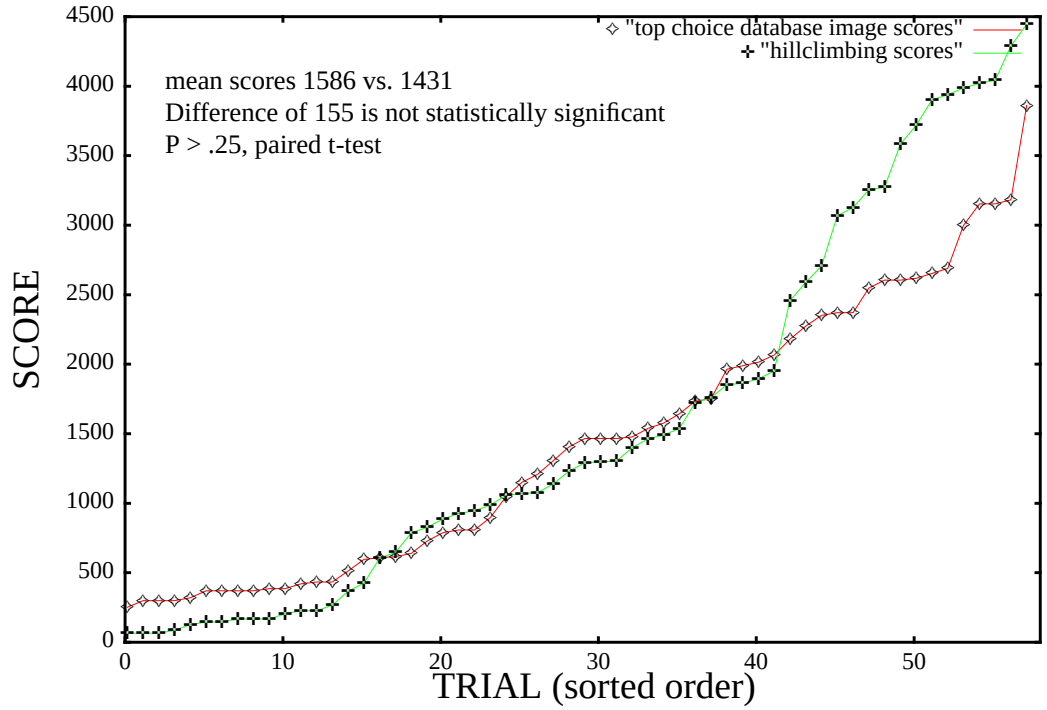

FIGURE 20. Human Trials: Random-Set vs. Hill-Climbing. This graph displays the 58 human trial scores from the final study in best to worst order (left to right) for the random-set strategy and the hill-climbing strategy. Hill-climbing starts out with the better (i.e., lower) scores, but ends up worse at the high end. Due to the bigger differences at the high end, random-set achieves the better average score (1431 vs. 1586 for hill-climbing), but it is not clear from this data whether either strategy is superior. Furthermore, the scores overall are so poor as to make both strategies quite unappealing.

5.7 Composites vs. Database Images Using Random-Set

While the scores for both random-set and hill-climbing were poor, we have not yet looked at how the use of Composites interacts with these strategies. The study results show immediately that the mean score of people's edited composite is dramatically better than the mean score of their first choice database image. Thus, using a composite constructed from images in the random set appears to be a better "version" of the random-

set strategy than using the top database choice out of the random set. Table 3 provides a comparison of the mean scores for subjects' top choice of database image (out of the 100 random images), their top choice random composite (recall that the study required choosing a favorite from among a set of ten composites constructed randomly from the top database choices), and their final edited composite. This table shows combined results for all targets from both the Pilot Study and the Final Study, as well as results from both studies on a per target basis. The table shows both the mean scores and a modified mean in which the scores above 1000 are all limited to 1000. This modified mean is a measure of the average search score assuming that people will lose patience and quit if the score exceeds 1000. Thus the number of such failures is also given in the table.

In all but one instance (Target G), the composite score is better than the score of the top choice database image. For the combined results, the difference between the mean scores of the top choice database image and the final edited composite is a substantial 430 image inspections. It is also interesting to note that, for Target 1 and Target D, the random composite was even better than the edited composite, and, on average over all targets (and more often than not on a per target basis), the random composite is somewhat better than the top choice database image. This suggests that the random composite interface is useful, a topic that is discussed further in Chapter 6.

TABLE 3. Final Study and Pilot Study Mean Image Score Results. Mean image scores for top choice database image, top choice random composite, and final edited composite are given in the table below. In addition to the mean score, we provide the modified mean score, which is the mean with "failures" limited to a score of 1000. If we assume that people are unlikely to continue looking past 1000 images and define any score over 1000 as a failure, then it may be more useful to look at this modified mean together with the number of failures. The number of failures and number of trials being considered are given in the third and fourth columns respectively. In each case, the row with the best scores among the three possibilities is shown in bold-face. Over all targets (shown at the top), the edited composite has the best mean score.

	mean score	**modified score**	**failures**	**out of**
All Targets (Final & Pilot)				
top database choice	1240	730	43	80
top random composite	1190	675	40	80
edited composite	**810**	**546**	**26**	**80**
Target 1 (Pilot)				
top database choice	762	658	5	11
top random composite	**277**	**277**	**0**	**11**
edited composite	454	379	1	11
Target 2 (Pilot)				
top database choice	1238	729	5	11
top random composite	1030	692	6	11
edited composite	**713**	**475**	**3**	**11**
Target A (Final)				
top database choice	1253	854	6	8
top random composite	1085	695	4	8
edited composite	**584**	**501**	**2**	**8**
Target B (Final)				
top database choice	1554	694	5	8
top random composite	1667	797	6	8
edited composite	**1133**	**623**	**4**	**8**
Target C (Final)				
top database choice	1993	973	7	8
top random composite	1970	943	7	8
edited composite	**1279**	**728**	**4**	**8**
Target D (Final)				
top database choice	1016	681	3	9
top random composite	**703**	**504**	**3**	**9**
edited composite	766	627	4	9
Target E (Final)				
top database choice	1037	563	3	8
top random composite	1786	717	4	8
edited composite	**573**	**423**	**1**	**8**
Target F (Final)				
top database choice	852	671	3	9
top random composite	967	733	4	9
edited composite	**368**	**368**	**0**	**9**
Target G (Final)				
top database choice	**1435**	**793**	**6**	**8**
top random composite	1726	883	6	8
edited composite	1657	898	7	8

Figure 21 shows the benefit of composites in graphical form. In this figure the final study individual scores for both the composites and the top choice database image are graphed in best to worst order (score is on the y-axis). Viewed this way, it is clear that the composite scores are superior. The difference of 435 is statistically significant ($P < .001$).

It is also interesting to view these same scores in a paired fashion (i.e., a particular individual's composite score vs. his or her top choice database image score). We present that data graphically in Figure 22, which shows the difference in score between each

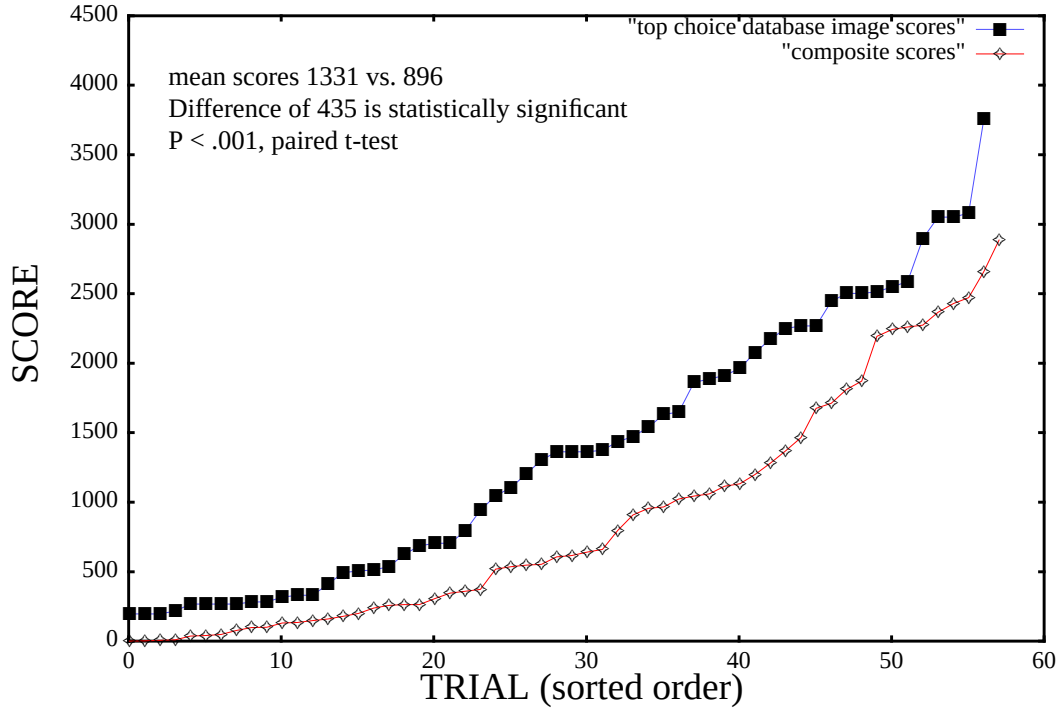

FIGURE 21. Final Study: Composite scores vs. top choice database image scores. Here we can see graphically the benefit of the composites. If we had to pick only one query image to use, a composite constructed out of parts from the 100 random images is a better choice than the subject's first choice of database images out of the 100 random ones. Note that both are clearly better than if the user had selected a query image at random. (An expected distribution of random selected query image scores could be graphed as a diagonal line from lower left to upper right.)

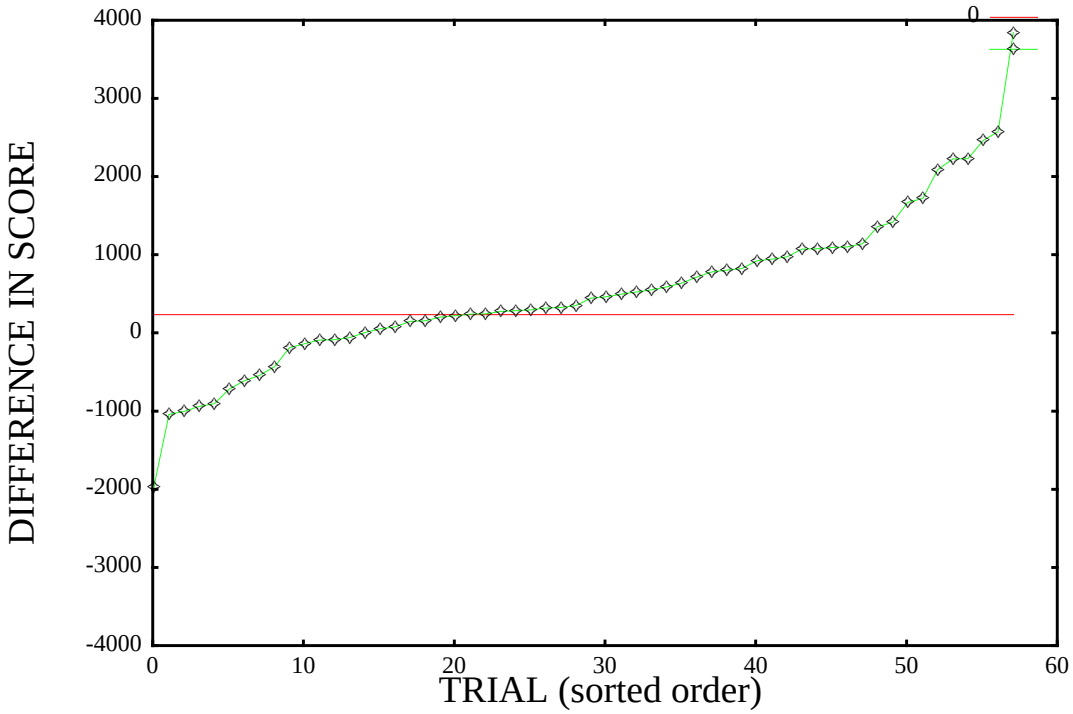

FIGURE 22. Individual differences between top choice database image and composite scores. Here we look at the score difference between each individual's top choice database image and the same person's best composite. The composites were better 70% of the time (i.e., 70% of the datapoints on this graph are greater than zero).

individual's top choice database image and the same person's composite (the score difference is on the y-axis). The composites were better 70% of the time (as indicated by the datapoints that are greater than zero on the graph).

To confirm that the average difference in score between the top choice database image and the final edited composite is not solely due to the case where the subject is looking at a photograph of the target throughout the experiment, we present the comparisons illustrated in Figure 23. These graphs, which split the data for both tasks into the "looking" and "not looking" cases, still show a statistically significant improvement in score for the composite, regardless of whether or not the subject was looking at the target while constructing it. It is also interesting to note that looking at the target during the experiment consistently produces better scores regardless of the task.

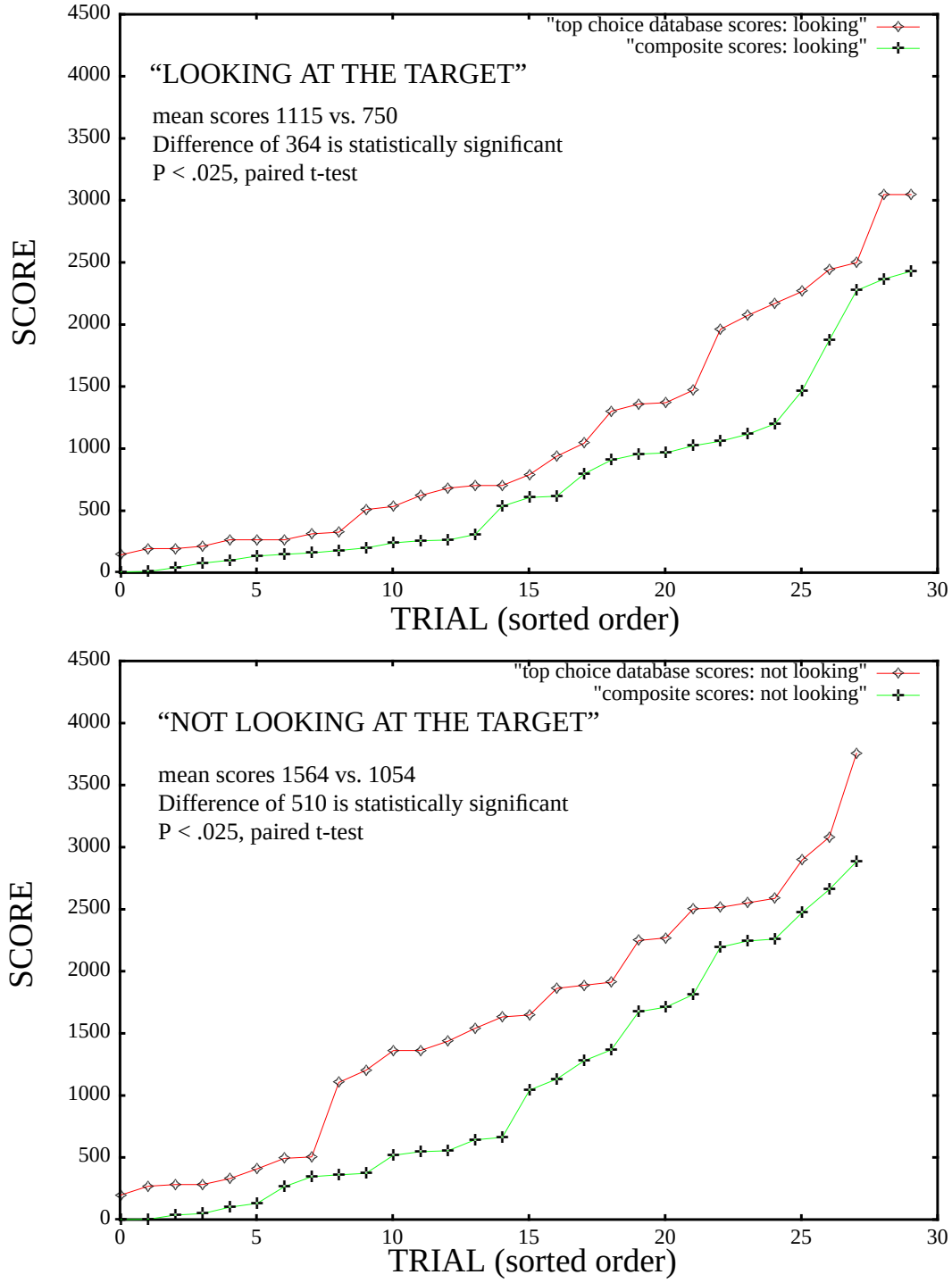

FIGURE 23. Not looking vs. looking at the target. The upper graph presents a comparison between the edited composite and the top choice database image scores with the subject “looking” at the target during the experiment. The lower graph shows the same comparison for the “not looking” case. In both cases there is a statistically significant difference between the scores, so it is clear that the improvement in score for the composites is not solely due to the “looking” case.

The success of the composites over the top choice database image is not yet convincing evidence that it is worth the time to create a composite. What about random-set strategies that use multiple query images? Recall that our study data includes computed image scores (per target) for each subject's top *five* database choices. We have already seen that 55% of the time the closest image in Eigenface space (of the 100) regularly shows up *somewhere* among the user's top five database choices. Perhaps one of the "parallel" variations on the random-set strategy using some set of the top database choices as queries would do as well or even better than using a composite. For example, if the closest image in Eigenface space were always to show up among the users' top five database choices out of the 100, then the strategy of searching in parallel the sorted lists based on these five choices would have an expected search score of 225 (plus the 100 initial inspections).[11] This is much better than the average image score of the composites created in the study. Indeed, for one of the two targets used in the Pilot Study, the closest or second closest image in Eigenface space was among the top five database choices for *all* eleven subjects. Ideally we want to be able to compare the optimal average search score among all variations of the random-set strategy that use one or more database images, to the optimal average search score among all strategies that use one or more of *both* the database images and the composites. Only such a comparison would permit us to determine if the composites are really necessary and, if so, how much benefit is derived from them.

11. Since 45 is the minimum expected score out of the 100 random selections from which the user is picking, we compute $5 \cdot 45$ (to account for the parallel search) to get 225. "Parallel" is perhaps a misnomer; we are really doing a breadth-first search,.

Unfortunately the huge number of possible strategy variations on random-set prohibits checking our user data for the average search scores associated with *all* of them. However, a simple characterization of most of the reasonable strategies does permit an exhaustive check of those. We define a "database image only" strategy as a triplet (H,D,I), where H specifies how many of the five database images to use as queries, D specifies how deep to look in the sorted lists based on these queries before going on to the next list, and I specifies how many such "breadth-first" iterations to perform before returning to look "depth-first" in the first list. For example, the strategy $(3, 40, 2)$ sorts the database for each of the top 3 (out of 5) database images, looks 40 images deep in each of the sorted lists, and then repeats this a 2nd time looking at the next 40 images in each list. Finally, if the target image has still not been found, it goes back to searching the remainder of the first sorted list, and keeps going until the target is found (thus we use the first choice database image as a "fallback"). We assume there is no reason to violate the user's ordering of the five images, so we exclude strategies that use the second image before the first, etc. Likewise, we exclude seemingly random tactics such as looking at image 200 in the first list, then image 46 in the second list, etc. We also assume there is no need to look at all possible values for D. Instead, we look only at multiples of 20 (one screenful of images) for the value of D. (Actually, we use 1, 21, 41, etc., so that pure breadth-first search [D=1] is included.) Finally, we make the assumption that 1000 is a limit on the search score for a strategy, since any more than that would likely tax a user's patience beyond its limit. A search that does not succeed in under 1000 image inspections is simply tabulated as a failure and included with a search score of 1000.

The above definition of a strategy does not yet include composites. To include them, we need only to change the definition into a quintuplet (H, D, I, P_1, P_2), where H now indicates how many of the seven images (the original five, plus the two composites) are used, D, and I are defined as before, and P_1 and P_2 specify the position of the composites in the image set. For example, the strategy $(3, 40, 1, 1, 0)$ places the random and edited composites in positions 1 and 0 respectively, thus bumping the database images down to positions 2 through 6. This strategy searches 40 images deep in each of the three lists associated with the edited composite, the random composite, and the top database image, in that order. If that fails, the search continues in the remainder of the list associated with the edited composite.[12] In this case, the "fallback" image is the edited composite. In general, for these strategies, the "fallback" image may be either the top choice database image, the random composite, or the edited composite. The set of strategies included in this definition is small enough that an exhaustive search can be performed on all of them, calculating the average search score of each from the raw user data collected in the study. For each target, we calculated the average search score over all subjects for each possible strategy included in our definition.

Table 4 gives the results of this exhaustive search. The shaded areas at the bottom of the table give combined results (i.e., the data from multiple targets is considered together) for the two targets used in the Pilot Study (1-2), the seven targets used in the Final Study (A-G), and all targets from both studies (1-2, A-G). The white area at the top gives results on a per target basis. We report here only average scores for the optimal strategies. The

12. When referring to a strategy we use x to mean "don't care", e.g., (1, x, x, x, 0) indicates the strategy that uses only one query image, the composite, which is placed in position 0.

complete raw image score data from which these averages are calculated is available in Appendix A. Initially we optimized only for mean score (where optimal was considered the *lowest* mean score). However, since it was possible to get a slightly worse mean score with a lower failure rate, we checked to see if any such situations existed. In two cases the mean score was worse by only 2 or 3 image inspections, whereas the failure rate was lower. In these two cases, we report the strategy with the slightly worse score, but lower failure rate.[13]

Many things are noteworthy about these results. First of all, it is abundantly clear that the strategies involving the composites are more effective than strategies using only database images. We now have a much more definitive answer about whether it is worthwhile to bother with the composites. With only one exception (Target G), strategies involving the composites (usually the edited composite rather than the random one) have significantly lower mean search scores and, in many cases, much lower failure rates, as well. This was true for both the per-target results and the combined results.

From the Pilot Study data, we initially concluded that the best random-set strategies were those that used a combination of both database images and composites as queries. We came to this conclusion because both the per-target results for Targets 1 and 2 as well as the combined results showed optimal strategies that used one of the composites as the primary query image, but also made use of some of the database images. We reported this conclusion in a paper on the Pilot Study results [1][2]. However, after collecting all the additional data for the Final Study, we came to a different conclusion. From the complete

13. When looking at the scores reported in this table, it is important to remember that they do not include the 100 inspections of the initial random set, so the actual search score is higher by 100 in all cases. Also, failures (i.e., search scores of 1000 or over) are averaged into the mean score as scores of 1000

TABLE 4. Optimal Random-Set Strategies: Final Study and Pilot Study Results. This table shows the optimal random-set strategies using only database images and using both database images and composites. The optimal strategy is determined by an exhaustive search over strategies represented by our strategy definition. The modified mean score given for each optimal strategy is the mean with "failures" limited to a score of 1000. If we assume that people are unlikely to continue looking past 1000 images and define any individual strategy score over 1000 as a failure, then it is more useful to look at this modified mean together with the number of failures. The number of failures and number of trials being considered are given in the third and fourth columns respectively. The optimal strategy is shown calculated on a per-target basis (unshaded) and calculated over all targets in the pilot study, the final study, and both studies combined (shaded). In all but one instance (target G), the optimal strategy using composites is superior to the optimal strategy using only the database images. Thus even if one considers random-set strategies that employ multiple query images (i.e., using up to five of the top database choices), a strategy employing a composite is likely to produce a better score.

	strategy	**mod. mean score**	**failures**	**out of**
Target 1: Database only	(4, 41, 4)	223	none	11
Target 1: Database + Composites	(6, 41, 1, 0, 2)	160	none	11
Target 2: Database only	(5, 1, 96)	580	5	11
Target 2: Database + Composites	(6, 61, 1, 6, 0)	382	2	11
Target A: Database only	(5, 121,1)	775	4	11
Target A: Database + Composites	(1, x, x, x, 0)	500	2	8
Target B: Database only	(2, 321, 1)	648	4	8
Target B: Database + Composites	(3, 1, 47, 1, 2)	568	4	8
Target C: Database only	(3, 261, 1)	878	5	8
Target C: Database + Composites	(1, x, x, x, 0)	727	4	8
Target D: Database only	(2, 401, 1)	655	3	9
Target D: Database + Composites	(2, 341, 1, 0, 2)	503	3	9
Target E: Database only	(2, 261, 1)	502	2	8
Target E: Database + Composites	(2, 261, 1, 2, 0)	370	2	8
Target F: Database only	(4, 1, 31)	383	2	9
Target F: Database + Composites	(5, 1, 31, 5, 0)	302	none	9
Target G: Database only	(4, 61, 1)	707	5	8
Target G: Database + Composites	(4, 61, 1, 4, 5)	707	5	8
Targets 1-2: Database only	(5, 41, 4)	423	5	22
Targets 1-2: Database + Composites	(6, 61, 1, 6, 0)	313	4	22
All Targets: Database only	(4, 41, 1)	664	40	80
All Targets: Database + Composites	(1, x, x, x, 0)	546	26	80
TargetsA-G: Database only	(3, 321, 1)	716	27	58
TargetsA-G: Database + Composites	(1, x, x, x, 0)	591	22	58

data set, it appears that *the simple strategy of using the edited composite as the sole query image is more successful overall than any other random-set strategy.* While strategies fitted to individual target results often made use of the database choices as well as the composites, the optimal strategy over all targets used only the edited composite. These results give a clear indication that the use of composites provides an advantage over restricting users to database images for their queries. In virtually all cases, strategies that include composites enable a user to locate a target face in fewer image inspections and with fewer failures.

5.8 Random-Set with Composites vs. Hill-Climbing w/out Composites

In our earlier analysis of the study results, we saw that without the use of composites the random-set and hill-climbing strategies appear to be comparable in search score. Since the random-set strategy is greatly improved with the use of a single composite, it now comes as no surprise that the random-set strategy of constructing a single composite as the sole query image works better than a strict hill-climbing strategy using only the database images as queries. The following graphs show a direct comparison of these two strategies as performed by our study subjects. At the bottom of Figure 24 on page 107 is a graph comparing the successful hill-climbing scores (i.e., those cases in which subjects actually found the target within ~1000 image inspections) to the best composite strategy scores (i.e., those that were also under 1000). The composite strategy score is computed as 100 (for the 100 initial inspections required to construct the composite) plus the image score of the (edited) composite. The scores in each case are shown in sorted order distributed evenly along the x-axis, from best on the left to worst on the right, and with the score on the y-axis. At the low end of the graph, the hill-climbing scores are slightly better. Here we

see that the 100 initial inspections required to construct the composite is an additional up front effort that has a slight cost compared to the small percentage of hill-climbing cases in which the subject was able to find the target quickly. Nonetheless, overall this up-front cost is worthwhile and the difference between the two strategies is overwhelming, with the composite strategy overtaking hill-climbing fairly quickly. Considering only the best 24 scores in each case, the mean score for hill-climbing is 390 vs. a mean score of 264 for the composite strategy, a difference of 126. Out of the 58 trials, 33 (57%) had a random-set composite strategy score under 1000, while for the hill-climbing strategy, only 24 (41%) had a score under 1000.

The 100 image inspections required at the outset to construct the composite may be qualitatively different from the image inspections required during an actual search for the target. For this reason, we also provide the graph at the top of Figure 24 which shows composite *image scores* (i.e., without the added 100) compared to the hill-climbing scores. Viewed this way, we can see that 36 (62%) of the composites had scores under 1000 vs. 24 (41%) of the hill-climbing scores.

Since we are looking at the best set of scores for both strategies, we cannot do a paired test on these data points. We can do an unpaired t-test, examining the difference between the two means of the best data points in each case, comparing the 24 successful hill-climbing scores (i.e., those trials in which the subject actually found the target) to the best 24 composite [plus100] scores. This shows the difference in mean (of 126 image

inspections) to be statistically significant ($P < 0.05$, t_{24}). However, this is a relatively small sample set. It would be nice to be able to make a paired comparison using all 58 trials. The following graph (Figure 25) provides such a comparison.

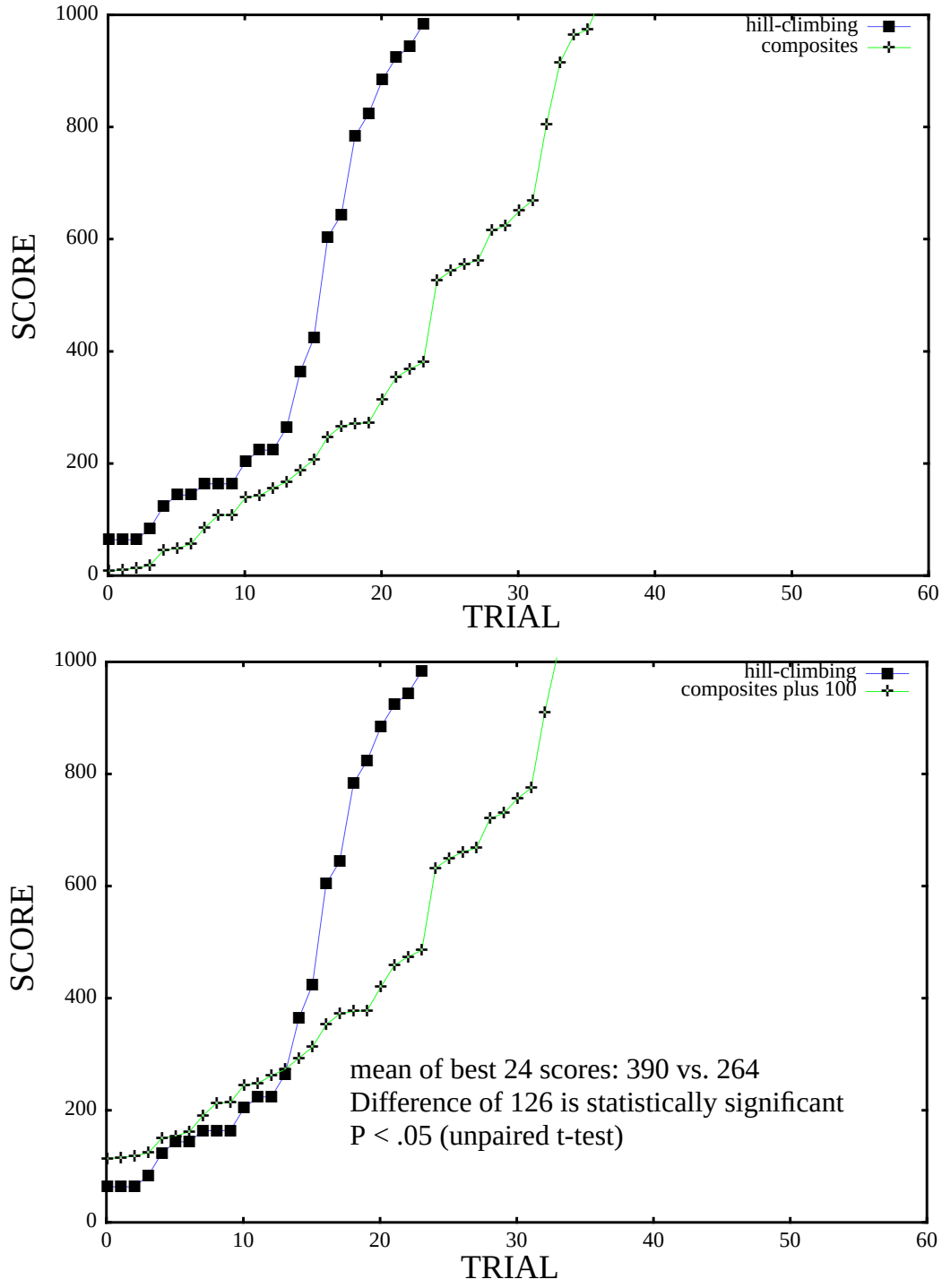

FIGURE 24. Comparison of hill-climbing and composite scores. Bottom: composite strategy scores (i.e., the composite image score plus the initial 100 image inspections) vs. scores of successful hill-climbing attempts. The composite strategy appears better, except at the low end. Top: composite image scores (i.e., without the initial 100 inspections added) vs. scores of successful hill-climbing attempts. In both graphs, only scores under 1000 are shown.

We can extend the graph at the bottom of Figure 24 to include all trials rather than just the best 24 by using the modified definition of hill-climbing, in which the scores for subjects who failed to find the target are set to 1000 plus the score of the last query image chosen. (Recall that subjects were asked to try hill-climbing until their image score reached about 1000). In this version of the strategy, we give strict hill-climbing a solid chance to succeed before reverting to a non-iterative approach. The obvious hope is that the earlier hill-climbing effort will have enabled people to reach a good final query image. Figure 25 below shows this comparison graphically, again with the scores for the two strategies sorted in best to worst order from left to right.

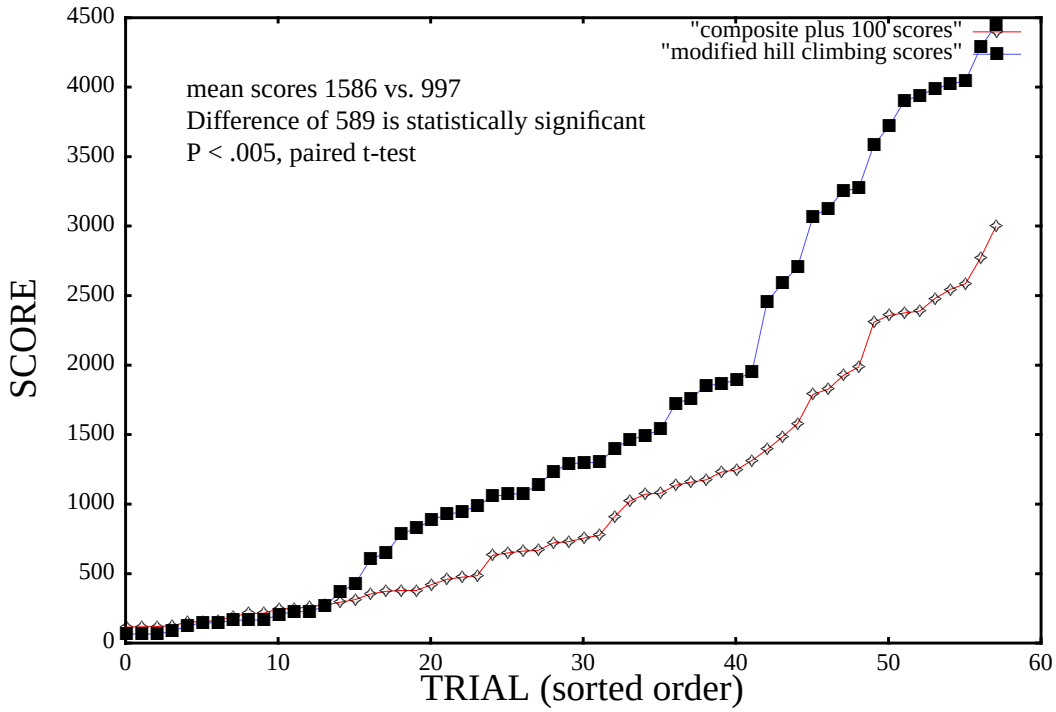

FIGURE 25. **Modified Hill-Climbing vs. Random-Set with a single composite.** Here we estimate the unknown scores for the hill-climbing "failures" —we take 1000 plus the score for the last query image chosen. We compare it to the actual composite scores (with the extra 100 included). Both sets of scores are sorted in best to worst order along the x-axis, with the search score given on the y-axis. Hill-climbing (the upper curve) is inferior, except at the low end, where the two are fairly comparable.

Doing the comparison with the modified hill-climbing strategy enables us to pair the sample points. The mean score for modified hill-climbing is 1586 and the mean composite-plus-100 score is 997.[14] A paired t-test (t_{30}) shows the difference of 589 to be statistically significant with $(p < 0.005)$. Figure 26 shows the paired differences between the two scores for each individual.

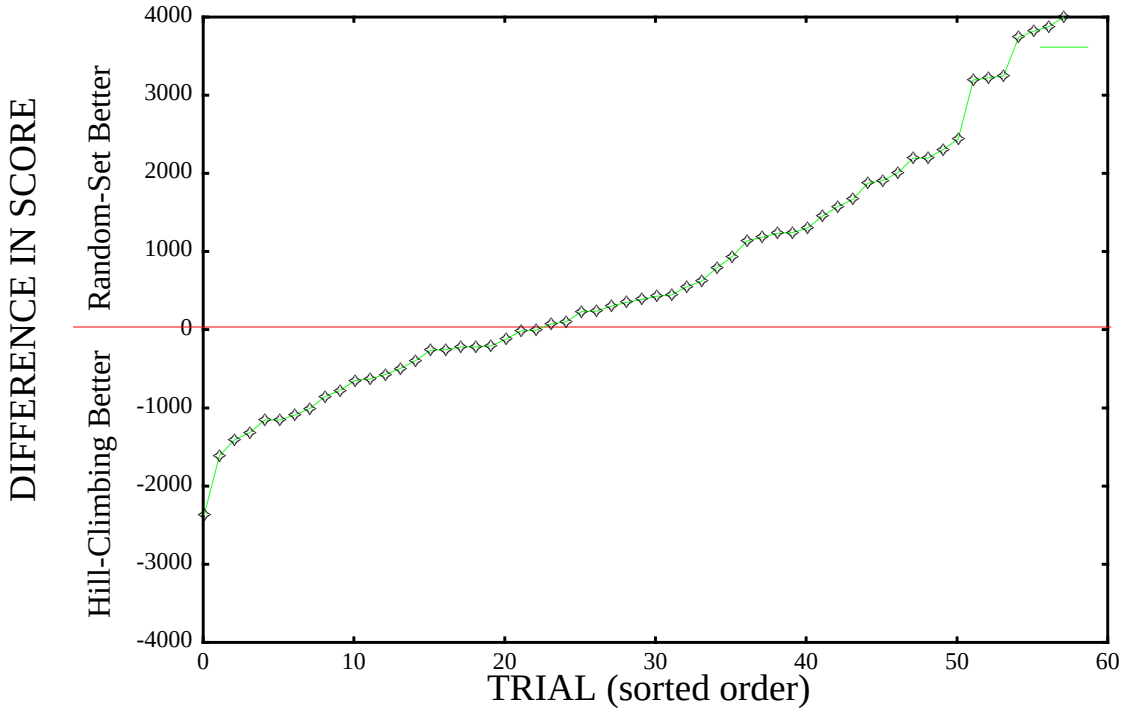

FIGURE 26. Differences between modified hill-climbing scores and composite(plus100) scores. These are paired differences, showing for each individual the difference in the search scores for the two strategies. Simply using the composite was the better strategy 59% of the time (the data points above zero).

14. The reason this is different from the composite mean reported in Table 3 is that this mean includes only the data from the Final Study, whereas the mean in Table 3 includes both Pilot and Final Study results. Also, it includes the 100 extra image inspections needed to construct the composite from the random images.

5.9 “Looking” vs. “Not Looking” at the Target

In the following graphs, we examine the impact of working from memory vs. working from an on-screen photograph of the target. As one might expect (since it simulates a photographic memory), working from a photograph seems to be helpful in terms of reducing average strategy scores, regardless of the strategy. This in itself is an interesting result; it indicates that Eigenfaces is doing something useful. A subject who has a “perfect memory” of the target face is presumably better at picking or creating faces that are perceptually close to it. If this in turn produces better strategy scores, it must be that the Eigenface metric also finds these perceptually closer images to be closer to the target. We saw this effect in Figure 23 as well, when we compared the composite scores to the those for the top choice database image (out of the 100 random faces). Scores with the subject looking at the target were better, regardless of the task.

For the strategy of creating a composite out of 100 random faces, the data show a difference between “looking” and “not looking” (Figure 27, upper graph), but it is significant only at the 10% confidence level. Likewise, for hill-climbing there is a difference between “looking” and “not looking” (Figure 27, lower graph), but it is significant only at the 25% confidence level. It is interesting to note that in this latter case the difference appears only when we look at scores over 1000, and these are the study trials in which the subject never actually found the target. This suggests that for the more successful hill-climbing trials, it did not matter whether or not the subject could see the target throughout the experiment. To a lesser degree, the same claim might be made about

the composite strategy (i.e., the curves in the Figure 27 upper graph are closer at the low end). Note that we could not evaluate either set of differences with a paired t-test because no single subject worked with the same target while both looking and not looking at it.

On the other hand, when we evaluate the difference in score between the two strategies and separate the results into two groups (a “looking” group and a “not looking” group), we can pair the trials. Figure 28 shows these results. This time we obtain a statistically significant difference in score regardless of whether or not the subject was working from memory of the target. The conclusion is the same as when we evaluated the results in aggregate (i.e., not separating the “looking” and “not looking” trials), namely that the composite strategy is better. We get a slightly tighter confidence result for the “looking” group (95% vs. 97.5%). If, in fact, the superiority of the composite strategy is more dramatic when a user has a particularly clear and accurate memory of the face, this would not be surprising. With a clear memory of the face, one ought to be able to use the composite approach to greater advantage.

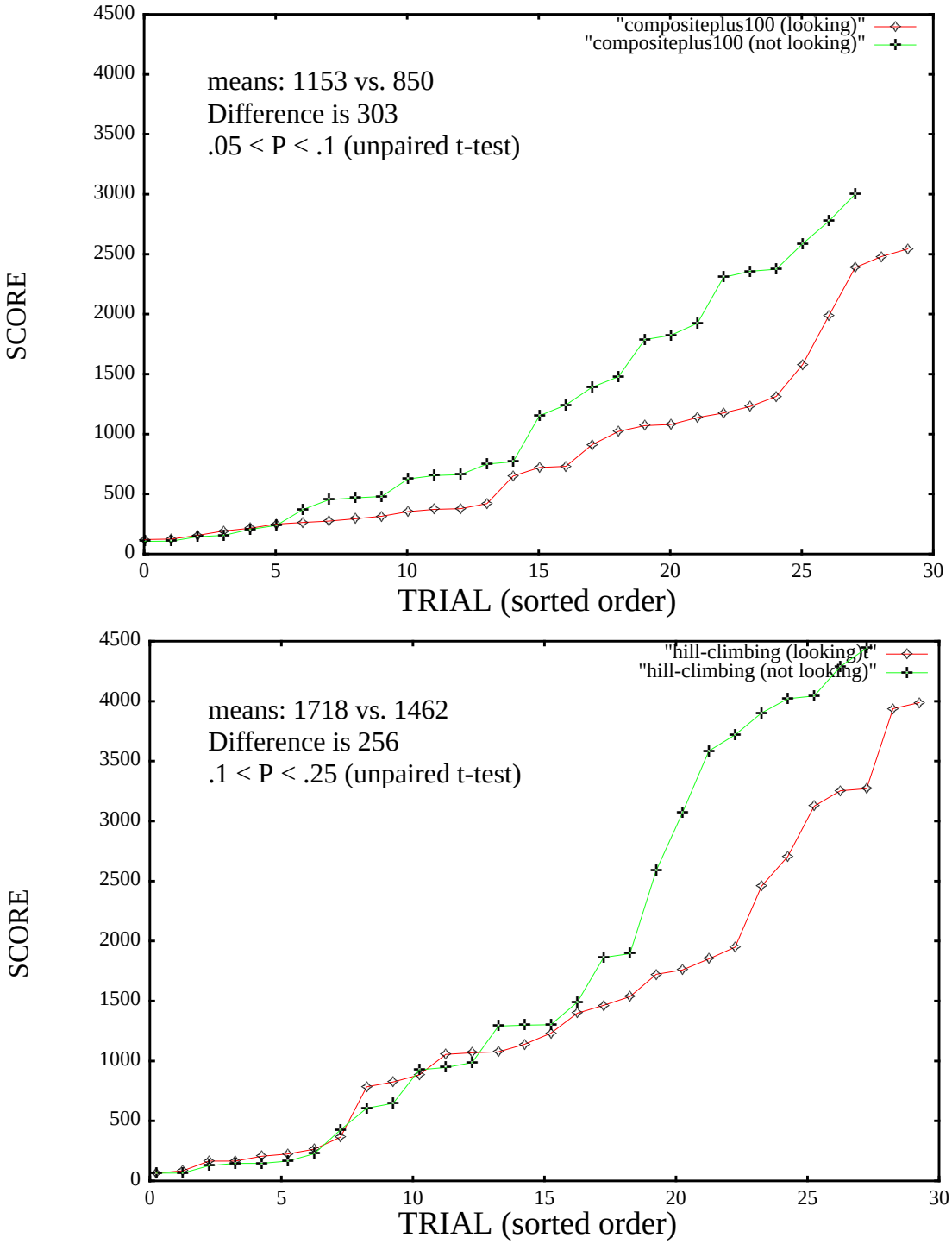

FIGURE 27. Composite Strategy "looking" vs. "not looking" (upper graph) and Hill-Climbing "looking" vs. "not looking" (lower graph). Average scores were better when the subject was looking at a photo of the target throughout the experiment regardless of the strategy. However, the effect of "looking" was more significant in the composite strategy. In the hill-climbing strategy, the effect was significant only at the high end.

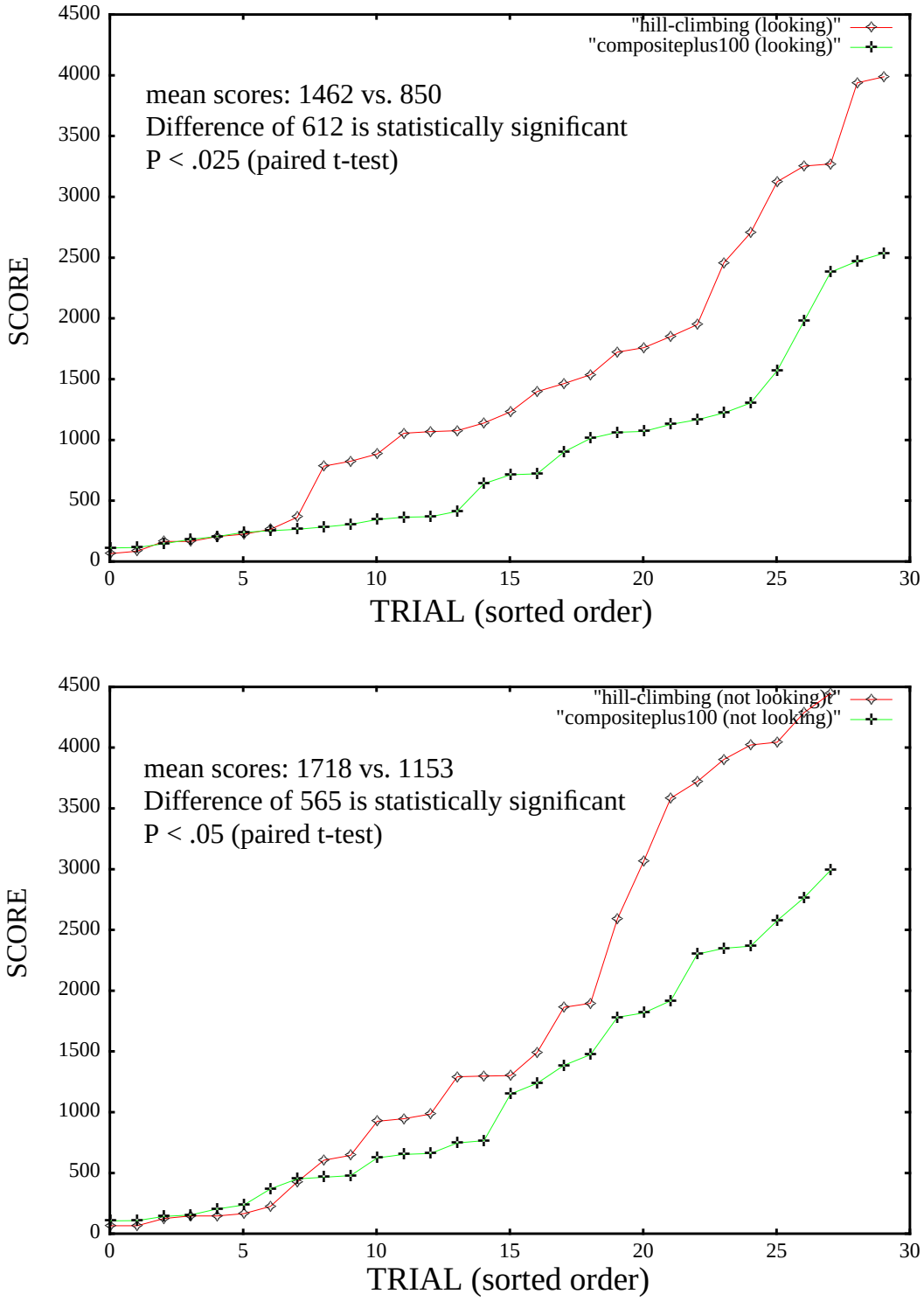

FIGURE 28. Hill-Climbing vs. Composite strategy, "looking" and "not looking." Here we compare the strategy of building a composite from 100 random faces to the modified hill-climbing strategy (i.e., for failure cases, we compute the score to be 1000 plus the score of the last query image chosen), but split the results into two sets, one in which subjects are working from memory of the target (i.e., "not looking", lower graph) and one in which they are working from a photo of the target (i.e., "looking", upper graph). Either way, the composite strategy is better.

5.10 Strategy Guidance to Users: Charting Effort to Expected Returns

Since the random-set composite strategy appears to be the most successful strategy we have seen thus far, it would be nice to get a better feel for just how useful this strategy is in practical terms. For example, a user might wonder what kind of effort (in terms of image inspections) is required to find a target in, say, a 4500 image database. The chart in Figure 29 expresses a kind of "effort to expected returns" ratio for this strategy. Assuming that our Final Study sample of 58 composites constructed by 30 different people is representative of the general public, this chart answers the question about how many image inspections are likely to be required to achieve a certain probability of success. The y-axis represents a number of image inspections and the x-axis represents a percentage of successful searches. To read the chart, select along the y-axis the maximum number, M, of image inspections you are willing to perform. The "max inspections" function (curve) shows the percentage of successes (given on the x-axis) for different values of M.

One might also wonder about the mean search time, or the mean search time for successes only. This information is given by the other two curves in the chart. For example, the chart indicates that if one wants a 68% success rate, one has to be willing to look at a maximum of 1000 images, with an expected search score of 547 and an expected search score among successes only of 328 (note that these numbers do not include the additional 100 inspections required to view the 100 random images when constructing the target). Although the chart doesn't account for skill differences among people in their ability to construct a good composite, it nonetheless gives a concrete, quantitative indication of the effectiveness of the strategy across a representative group of people.

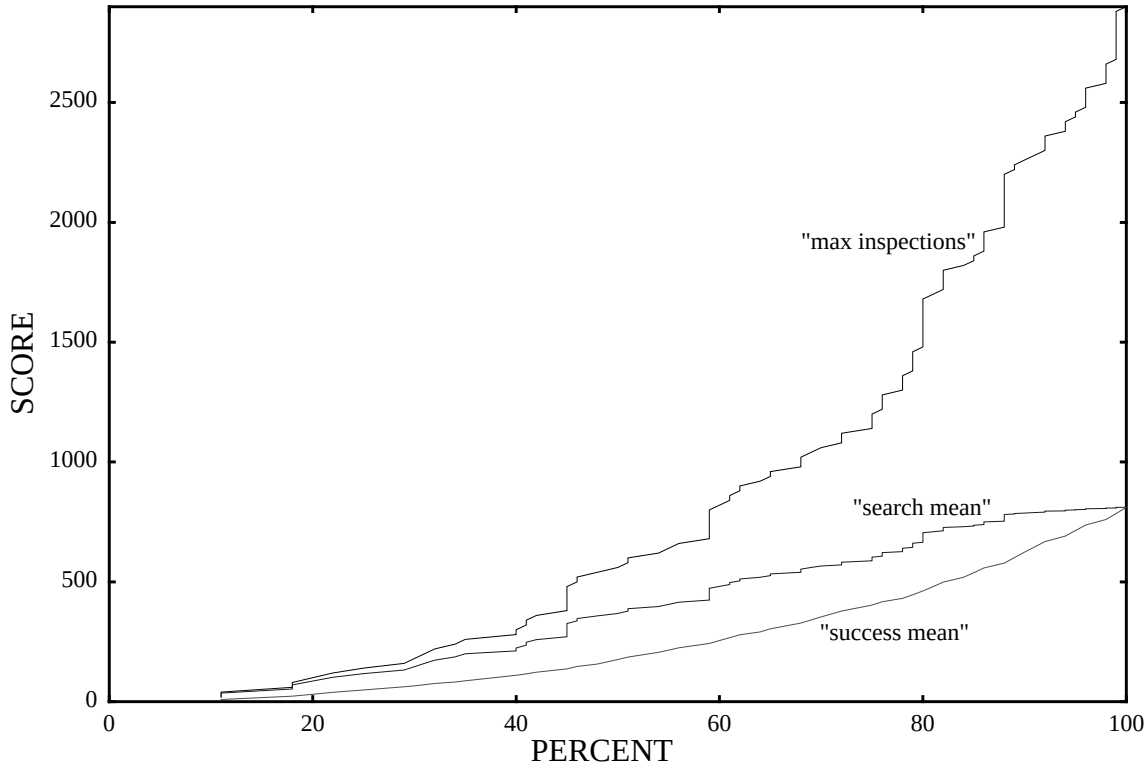

FIGURE 29. Composite Strategy Guidance Chart. For the simple composite strategy, the uppermost curve of this chart shows on the x-axis the percentage of successes (out of 58 human trials) if users limited their number of image inspections to the number on the y-axis (the max inspections curve). Below this is a curve representing on the y-axis the mean number of image inspections for a search having the probability of success of the number on the x-axis (search mean curve). The lowest curve represents the mean number of inspections for the successes only (success mean curve).

5.11 Why Is Hill-Climbing Not Working?

The obvious question to ask at this point is why hill-climbing, which appeared to be such an excellent strategy when used with a “perfect” metric, does not appear to be working well under more realistic conditions. Clearly the real-life metric interacts poorly with hill-climbing. Some significant portion of the time users are mistakenly guiding the search down the slope instead of up, a situation that wreaks havoc. If a user makes a

“mistake,” the consequence is time wasted cycling around in an unproductive region of the database. Since hill-climbing requires the user to make many decisions about choice of query image, the combined negative impact of the “bad” choices can be very costly.

Perhaps there is some useful way to modify the hill-climbing strategy to take advantage of some of the knowledge we now have. For example, we certainly would like to know how well hill-climbing does in combination with the use of composites. It is possible for users to construct a composite in an iterative fashion rather than from a single random set. Interim composites could be used to search the data, and the results of interim searches could be used to improve the developing composite. We have seen that the use of composites greatly improved the performance of the random-set strategy. Our current study does not provide any real data on this topic (of hill-climbing in combination with composites) other than to suggest that the composites are useful. Answering this question requires an additional study and further discussion of it is left for Section 7.2 on future work.

However, there may be other ways in which we can analyze/improve the hill-climbing strategy. We have seen that hill-climbing either does well in about 200 image inspections or, after that point, it does much worse than the random-set strategy with composites (see Figure 24). This suggests that it might be worthwhile to try hill-climbing early on, but to give up on it much sooner than at the 1000 image inspection point. The problem seems to be that people don’t know at what query image to stop iterating. In general, the decision

about how far to look at each iteration is a tricky one. We noted that if our study subjects failed to see the target right away in one or two pages of faces after initiating a query, they often got discouraged and would opt to try a new query image rather than search further.

From the study data we can compute revised scores for a hypothetical situation in which people are clairvoyant and know exactly at what point to stop iterating and search the current query list sequentially. Figure 30 illustrates the results of such assumed clairvoyance with a graph comparing the best possible hill-climbing scores to the best possible composite (plus 100) scores. What is meant by this is that these are the scores as they would be if the subjects always knew the optimal point at which to stop iterating while hill-climbing, or to stop tinkering while constructing a composite. Note that in the case of hill-climbing this is not just a matter of stopping at the query image with the lowest score because a query image that is found earlier on but with a worse score may be a better choice than a query image found later with a better score. In the case of the composites, the clairvoyance is simply a matter of knowing at which stage the composite has reached its lowest score (it was not uncommon for people to tinker with their composites to the point where the score worsened rather than improved).

This is a hypothetical situation and real people do not have this kind of clairvoyance, so the superiority of hill-climbing in this comparison is not real. Nonetheless, it is interesting to see how dramatically the hill-climbing scores improve if people were to know when to quit iterating. This graph highlights a big problem with hill-climbing, namely, that people simply don’t know which query image to stop and stick with. The critical decision about whether to continue searching a query list or select a new query

image gets made over and over again in hill-climbing, and users are ill-equipped to make it wisely. Apparently, the feedback they are getting from the algorithm regarding whether or not they are on the right track (i.e., the look of the other faces in the query list) is not sufficient to guide them well in making these critical decisions. On the other hand, with the random-set composite strategy this decision is made only once, so the effect of a bad decision is not compounded as it is with hill-climbing. However, with hill-climbing the user gets to cover more ground, touching upon a wider range of database regions, so the odds of finding a close query image are greatly increased. This leads us to be optimistic about a strategy that combines hill-climbing and composites. For example, perhaps one should use hill-climbing early on, meanwhile gathering images along the way as possible contributors to a composite. If hill-climbing fails within a few hundred image inspections, a composite can then be constructed from the images collected and used as the final query image. This puts off the extra work of constructing the composite unless and until it becomes necessary. Alternatively, the composite could be constructed as an integral part of the hill-climbing process itself, with the iterations stopping at some point fairly early on. The determination about which, if any, of these integrated strategies is best is left for another study, but ample evidence exists from this work to suggest that such strategies may work well.

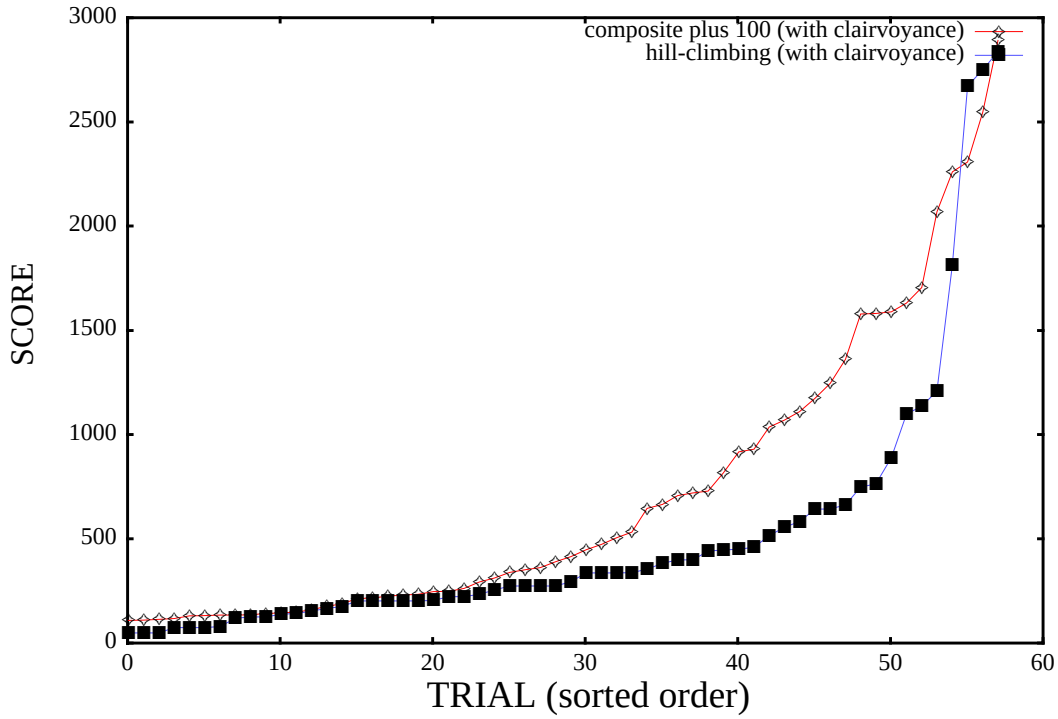

FIGURE 30. The lower curve shows the best possible score for each hill-climbing attempt (i.e., the score if the subject had known the optimal point in the search at which to stop iterating). We do something similar for the composite scores (upper curve) —as the composite is being constructed, we compute the score as if the subject had stopped editing at the best interim composite (i.e., the one with the lowest score). In this comparison, hill-climbing looks superior. This may be because the problem with hill-climbing is precisely that the user doesn't know at what point to stop iterating. In addition, the user gets to travel more ground (i.e., try out more regions of the database) while choosing candidate query images than was possible during the composite construction process.

5.12 Summary and Discussion

Human studies are clearly essential, since it is only through such studies that one can see how the similarity metric interacts with the choice of search strategy. We saw in the simulations that the hill-climbing strategy was superior in the presence of a metric that correlates perfectly with the "human" one. The simulations showed that hill-climbing and

random-set, in principle, have the potential to reduce mean search scores to about 2% and 4% of the database respectively. But the human study showed that this superiority of hill-climbing is not sustained when the Eigenface metric has to be used in conjunction with human similarity criteria. In fact, when these strategies (without the use of composites) were employed by real people, random-set was better, but neither worked well, yielding mean search scores of only about 32% (random-set) and 35% (hill-climbing) of the database. Although this is an improvement over the 50% required for sequential search, it is not sufficient to be practical for large databases.

However, our study showed conclusively that the use of composites as queries is an effective means of improving mean search scores. The strategy of constructing a single composite out of a random set appears to be the optimal random-set strategy, reducing the mean search score to about 20% of the database size.[15] This approach begins to cross the threshold into a truly practical system, although there is still much room for improvement.

We conjectured that the problem with hill-climbing under real-life conditions is that the limited level of correlation with the "human" metric too often causes the user mistakenly to guide the search "down the slope" into unproductive regions of the database. Because the user must make a guiding decision at every iteration, bad decisions are compounded and much effort can be wasted. Even when the search really is in the right region of the database, there often is insufficient feedback for the user to realize this and

15. The mean edited composite score over all targets was reported in Table 3 on page 95 to be 810. We add 100 to this for the 100 random inspections and then divide the result by 4500 (the size of the database) to get 20%

make a wise decision about when to stop iterating. Given these problems, it seems likely that a strategy employing some form of hill-climbing early on, but that resorts quickly to a non-iterative composite approach will yield even better results.

In discussing strategy it is always the case that a different similarity metric might produce different results. While our study focused on identifying successful strategies in a system employing full face Eigenfaces, for systems that employ other (possibly better) mechanisms for determining similarity between images, the answers may be different. Our suspicion is that our strategy conclusions for the Eigenface metric will apply to any of the metrics currently proposed in the research community. All metrics, like Eigenfaces, are severely hampered by the lack of agreement about similarity judgements among people and the negative impact of this phenomenon may swamp any small improvements that can be gained by another metric. Nonetheless, in view of the many possible metrics, variations, and parameters that can have an impact on performance, system designers will perhaps not be relieved of the task of testing their own particular metric's impact on search strategy. Should it be true, as some claim, that other metrics outperform Eigenfaces, the methods described in this chapter for evaluating and comparing various search strategies can be generally applied to other systems. They can be used both to determine the best search strategies for a given metric and to help distinguish between the many possible candidate metrics.

Chapter 6
Query Interface

6.1 Introduction

Previous chapters discussed how the similarity metric and choice of search strategy can be used to improve average search scores in a mug-shot search system. However, the scores obtained are still less than ideal. In this chapter, we explore the interface for formulating queries (e.g., composites) to see what kinds of interface features might have the potential to produce further improvements. It is especially interesting to look at how the integration of the search engine and the composite creation system gives rise to many intriguing potential interface components that would otherwise be impossible if these two subsystems were treated as completely separate independent units. In fact, even if one is not interested in searching a database, but rather is interested primarily in constructing a composite, integrating database search provides a convenient way of accessing an enormous palette of thousands of faces and face parts.

The composite construction interface used in our study was fairly rudimentary and, furthermore, subjects were not permitted to take full advantage of many features the system did have. We limited subjects because we did not want to confuse our strategy analysis with too many variables, and we certainly did not want to risk making the query interface so abundant with features that it became overcomplicated. (There is a critical trade-off in any interface between simplicity and added functionality.) Since subjects were not allowed to use all features, study data regarding the effectiveness of different features of the query interface is limited. However, we experimented informally with many

interface ideas and can offer our thoughts as well as some anecdotal and experimental evidence regarding the potential value of several interface components. We summarize critical design choices and outline possible advantages and disadvantages associated with each.

6.2 Random Composites

One aspect of the system on which we did gather some limited data is the interface for constructing random composites. The idea behind this interface is that people are sometimes better at assessing whole faces for similarity to a target face than they are at recalling individual facial features. The interface lets the user select several “parent” faces as a starting point, and the system quickly creates and displays sets of faces that randomly combine facial features (eyes, nose, mouth, forehead, chin, etc.) taken from the “parents.” Instead of having to recall and then manually edit individual features, the user can identify whole faces that bear some resemblance to the target and then peruse more sets of whole faces in which the various features are randomly shuffled. The user can repeatedly generate such sets of random composites until something useful turns up. The set of “parent” faces can be modified as desired on the fly. Any composite may be used as a query to conduct a database search, and the results of a search may also be used to add to or refine the set of “parent” faces. If a particular feature or features seem “just right”, those features can be fixed in the random composites, so that only the other features vary. Figure 31 shows examples of random composites generated from a small set of “parent” images. Figure 32 illustrates the fixed feature mechanism.

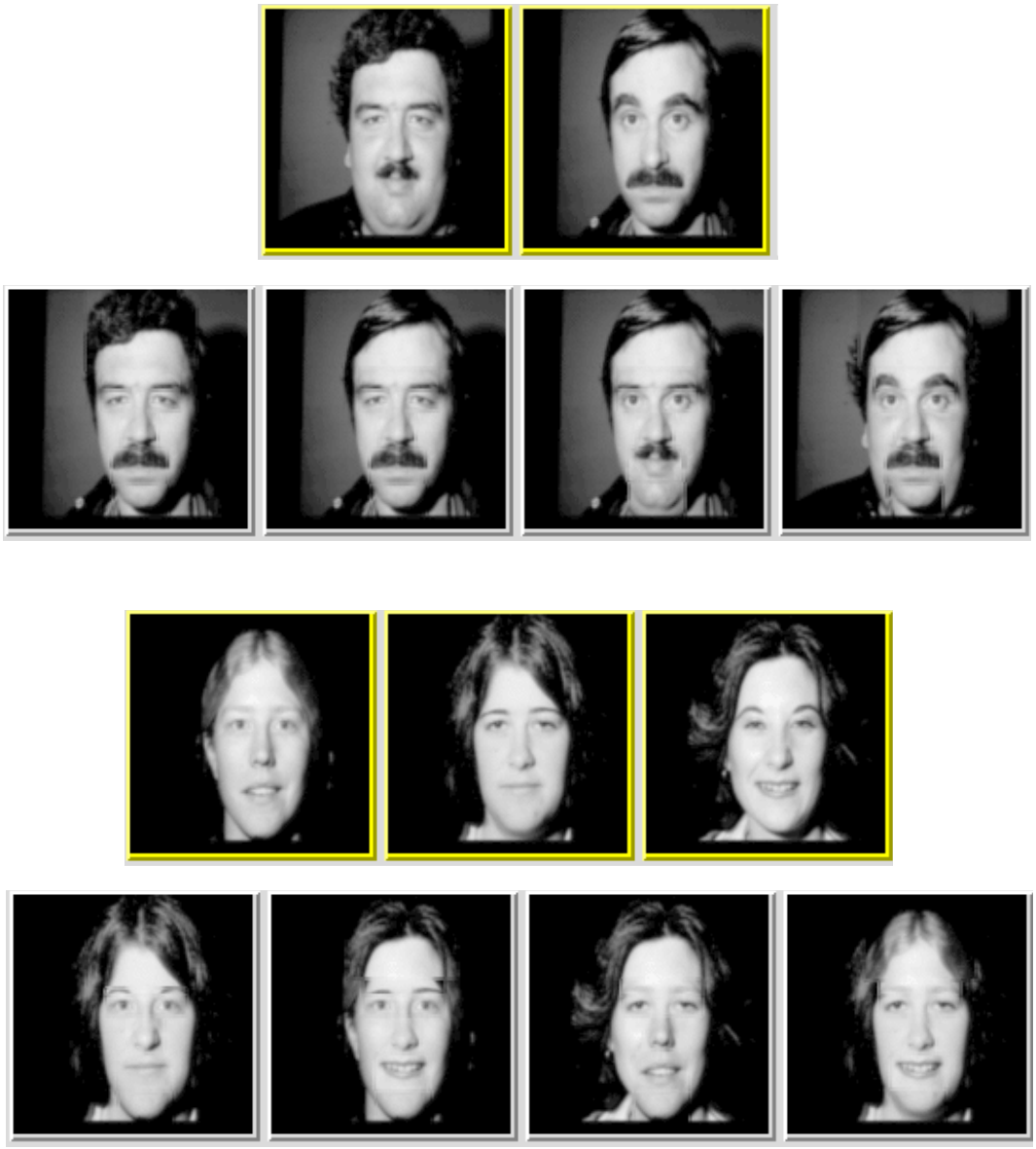

FIGURE 31. Random Composites. The top row shows two database faces that were used to generate the four random composites shown below them. Next (third row of faces) is a set of three database faces used to generate the four random composites shown below them (at the bottom). The user can generate as many random composites as desired and can also modify the number and contents of the group of "parent" faces as desired.

.

FIGURE 32. Random Composites With Fixed Features. The top row shows three database faces that were used to generate all the random composites shown below them. The first set of random composites (middle row) have the cheeks and forehead fixed from image B. The second set of composites (bottom row) have the eyes fixed from image A and the eyebrows fixed from image C.

The random composite interface is related to ideas used in the FacePrints composite construction tool. We suspect that the full blown genetic algorithm approach used in FacePrints (in which the user has to rate each generation of 30 composites for similarity to the target) may be unnecessary overkill. Still, isolated feature recall is sometimes difficult, and people may very well be better at employing a recognition-based approach. Even when the user can recall isolated features, the manual editing of individual features can be cumbersome and the ability to pick whole faces from sets of randomly generated composites may sometimes be faster and more effective, especially since manual editing

can always be used as well to gain more refined control over the results. In any case, a simpler interface than FacePrints (i.e., random composites but no genetic algorithm) is probably sufficient. The integration of the Eigenface search capability gives users an effective means to navigate the space of the database looking for potential "parent" faces or features. Note that the random composites are also related to the interface idea used in the SpotIt system [5] in which the user can interactively interpolate between up to three features (e.g., eyes), viewing the Eigenfeature reconstruction of the result. Here, too, a set of "parent" images are used to explore alternative images in the search space neighborhood of the parents.

Although we have no definitive statistical data on the value of the random composite interface, some of the study results indicate that it is useful. We were concerned about keeping image inspections to a minimum so that we could make a fair comparison of various strategies. Because of this, study subjects were restricted to looking at only 10 random composites, which were generated from their top five image selections from the database. Subjects were not given any choice about which of their five database choices were used for generating random composites. They also did not have access to the feature-fixing mechanism during the random composite experiment, and they were not allowed to edit manually even one of the random composite's features (at least not until they were asked to create a complete composite via manual editing). All these factors worked against the likelihood of successful use of random composites. Yet in spite of this, the top choice random composite (out of the 10 generated) still figured prominently in several of the "optimal" random-set strategies that were identified in Table 4 (in Chapter 5 on page 103). When considering the optimal strategy averaging search scores over all subjects and all

targets, only the manually edited composite was used. But for the optimal strategies on a per target basis (i.e., with search scores averaged over all subjects per target), the random composite was used in three out of nine cases (Target 1, Target B, and Target D). Recall that by definition, all these strategies (as defined in Chapter 5, Section 5.7) have a "fallback" image, the image in position zero, which is used as the final query image if the strategy otherwise fails. Presumably, the image used as the fallback is (on average) the best single query image among the three possibilities.[16] For five out of the nine targets, the final edited composite was the fallback, but for Target 1 and Target D, the random composite was the fallback. (For only one of the nine targets (G) a database image rather than a composite was used as the fallback.) The mean score of the random composites was also consistently slightly better than the top database choice (see Table 3 on page 95). And for Target 1 in the Pilot Study, the random composite had a mean score that was significantly better than the edited composite. Although none of this evidence is overwhelming, it does suggest that the random composites are useful. However, there may also be a potential negative effect due to degradation of the mental image as more and more sets of random composites are inspected.

6.3 Feature-Based Retrieval

One particularly interesting way in which the composite building interface and the search capability can be combined is with the use of Eigenfeatures, the method described in Chapter 2, in which PCA is applied to individual facial features. Using Eigenfeatures, one can search the data for individual facial features (such as noses, eyes, mouths,

16. The three possibilities for the fallback image are the first choice database image, the first choice of random composite, and the final edited composite.

foreheads, etc.) that resemble those belonging to a query image. Our tools permit feature based searching on any subset of the features (e.g., just the nose, or both the nose and the mouth, or the eyes, chin, and forehead all taken together, etc.). The user can easily set and reset these parameters at any time, changing them between queries as desired.

We were uncertain exactly how many of the feature vectors ought to be used in calculating distances. The more eigenvectors used, the more accurate the reconstruction of the original image. Reconstructions using different numbers of eigenvectors give a rough sense for how much information is represented by the leading N vectors. Figure 33 shows mouth reconstructions using 5, 10, 20, 25, 50, and 125 of the first eigenvectors (the original appears to the far right). Since 50 was clearly the maximum needed, the retrieval results shown in examples in this chapter use 50 vectors weighted equally, although this is probably more than is necessary.

FIGURE 33. Mouth Reconstructions Varying the Number of Vectors. These reconstructions show the image quality achieved using the first 5, 10, 20, 25, 50, and 125 of the mouth vectors. The original is at far right.

Figure 34 shows an example in which the composite in the upper left is used as a query to search for similar noses. The faces in the bottom row of this figure were found on the first page of the list of database images sorted by distance from this query nose. Each of the noses is tried out on the composite above it, making it easier to see how the noses are similar as well as how they are subtly different. Searches based on individual features can

help users find just the parts they need to construct the desired composite. Figure 35 shows another example using eyes as the query feature. In this example it is particularly difficult to recognize that the eyes are similar in the set of query results because the eyes appear so different in the context of the full face. This suggests that an interface that displays the results of a feature-based search on a single standard face (perhaps an "average" face) might be more effective than displaying features on the face to which they happen to belong. Figure 36 shows an example in which the *nose and mouth* are used as a *combined* feature for a search. Faces that were found via this search were used to construct the set of random composites shown in the top row (in which the eyes feature is fixed from yet another face). This figure illustrates how the various interface tools may be used in combination to produce a desired result.

FIGURE 34. Searching For a Nose. The bottom row of faces were identified via a search for similar noses, using the composite in the upper left as a query. The face whose nose this composite came from was found first (bottom left), and three other faces (to its right) were found with similar noses. Each nose was tried out on the composite above it, which makes it easier to see their similarity.

FIGURE 35. Refining Composites. The bottom row of faces is the result of a search on the eye feature only, using the composite in the upper left as a query. The database face whose eyes this composite used is first in the list, followed by other database images found by the metric whose eyes are similar. It is almost hard to tell that the eyes are all similar because they look so different in the context of these other faces. But when we place the four different sets of eyes onto the composite (each composite in the top row is immediately above the database face whose eyes it has), the similarities between these eyes as well as their subtle differences become more apparent. This technique can be used to refine a composite if, for example, one wanted narrow eyes, but with a subtly different look.

FIGURE 36. Combining Fixed Features, Feature-Based Search, and Random Composites. The upper row of faces were found via a search on the *nose and mouth* features combined using the upper left face as a query. These four faces were then used to generate the set of random composites in the bottom row. These random composites were generated with the eye feature fixed from yet another face (not shown). All of these various mechanisms can be combined to give the user a varied palette of composite construction tools.

6.4 Painting Tool

The Painting Tool permits users to paint a rudimentary hat, moustache, beard, etc. directly onto the composite. It also works well to alter the hairline, lengthen the chin, or make other adjustments to various gross image features. The paint color (i.e., the pixel intensity value) is selected by clicking directly on the desired color in any image. Figure 37 shows an example in which the painting tool is used first to add a moustache, then a beard, and finally some extra hair over the forehead. This figure also illustrates how each of these changes affects the results of searches in which each painted composite is used in turn as a query. Figure 38 shows an example in which the painting tool is used to create a rough sketch of a white hat. The query results include people wearing white hats and people with white hair. The painting tool is good for altering gross image features, but would probably be impractical for making subtle changes. Combining this type of QBIC-style [11] rough-sketch interface together with an interface designed specifically to capture the appearance of faces offers users the best of both worlds. Painted images can be used to locate quickly faces with desired features, and these features can, in turn, be used to produce a more refined version of the composite. We also found the painting tool to be a useful research aid because, as illustrated in Figure 39, we could use it to modify various image features and then see how these changes affected the image score. Thus it helped us to develop some intuition for what features were important to the Eigenface metric. As one might expect from a metric based on the full image, it is especially important to get the gross image features correct in a composite (more important than getting subtle details correct), and this is exactly the task for which the painting tool is most useful. Changes to the shape of the face, the forehead, or hairline are particularly simple to implement.

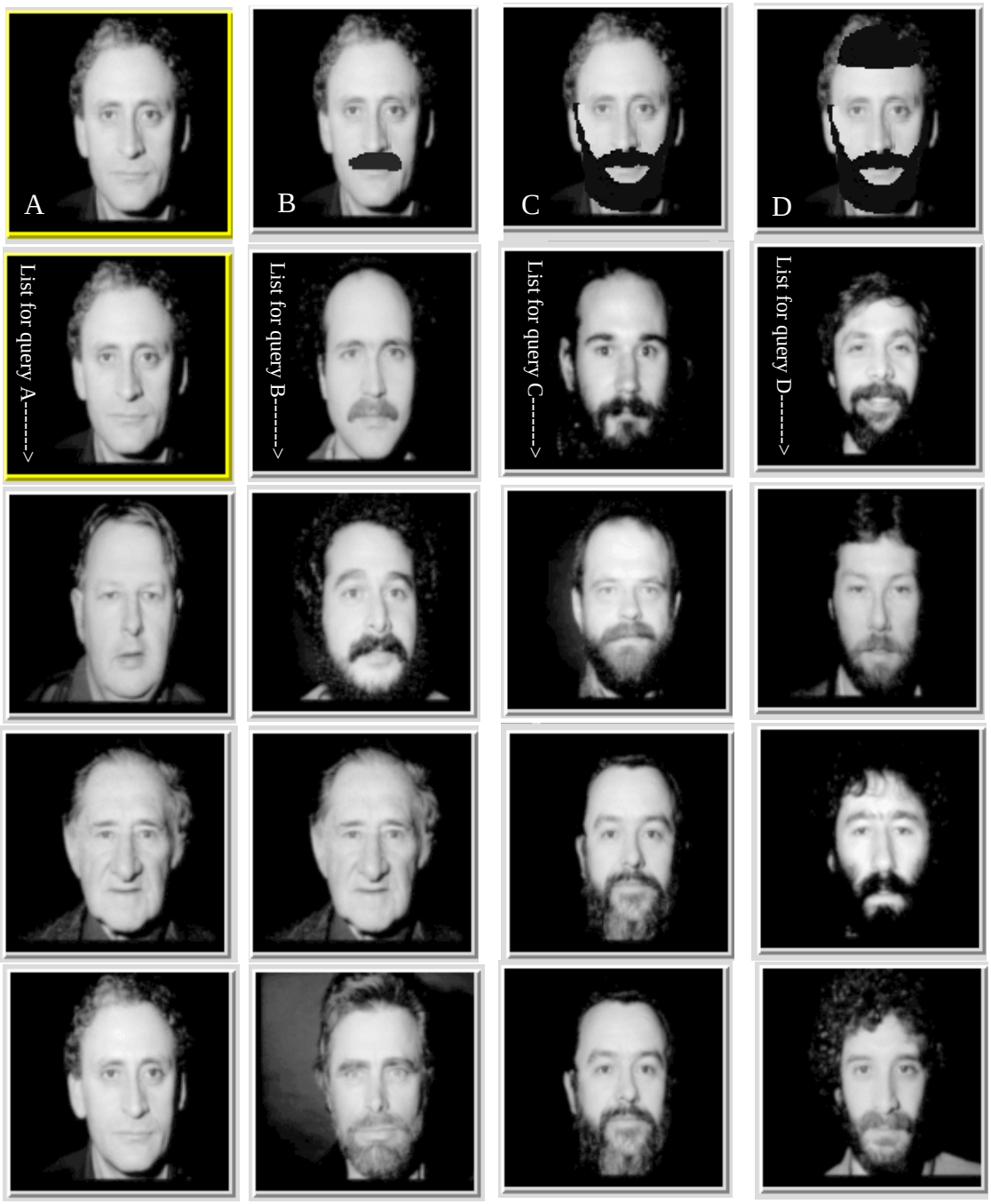

FIGURE 37. Painting on Composites. The top row of faces shows a progression of changes in facial and head hair created with the painting tool. Below each face in the progression is a column showing the first four faces in the list of database images sorted by distance from images A, B, C, and D, respectively. Rough sketching over faces using the painting tool is another way users can quickly modify their composite and locate related faces in the database. Features from these faces can, in turn, be used to modify the composite to obtain a more refined result.

FIGURE 38. Painting Tool Example 2. In this example a crude white hat is sketched on a database face and the resulting image used as a query. The first page of query results, shown below, turns up people with white hats as well as people with white hair.

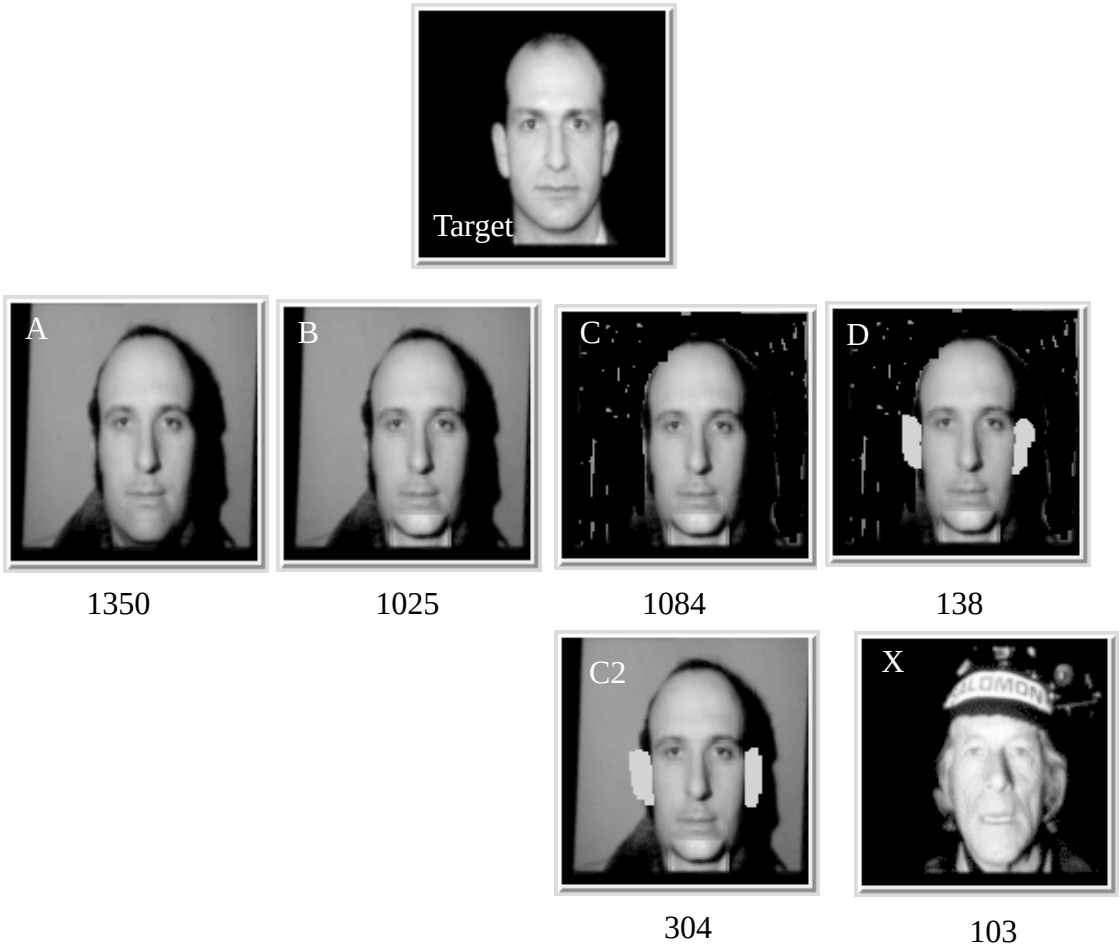

FIGURE 39. The Painting Tool As a Research Aid. In this example, the painting tool was used to help us better understand the behavior of the Eigenface metric. The top face is a target that was used in our study. Subjects in the study were unusually consistent in their belief that face "A" looked similar to the target. Yet its image score was high (1350), indicating that Eigenfaces did not agree with people. We wondered why and tried several changes, some with the painting tool, to determine which changes would most improve (i.e., lower) the score. (The score of each image is given below it.) First we shortened the chin (B), and this did improve the score somewhat. Then we speculated that perhaps the lighter background was the problem, so we painted in a darker one (C). Oddly the score got worse, indicating that it was not just a matter of background differences. Then we painted in some crude ears (D) and the score improved dramatically. We wondered whether the dark background was necessary at all, so we tried just the ears without the darker background (C2) and this did improve the score quite a bit, but not as much as when the darker background is included as well. Thus the painting tool was useful in helping us develop some intuition about what features were most important to our metric. In this case, it appears that the ears, especially as they contrast with the darker background, were a critical feature. This explains Eigenfaces' puzzling conclusion that image X was the closest image in Eigenface space to the target (from among the 100 random images used in the study). Apparently the light hair sticking out on the side of the face (against the dark background) was "perceived" by Eigenfaces to be similar to the ears on the target face.

6.5 Reconstructions vs. Photoshop-style Constructions

Do Eigenfeature-based reconstructions (as used in the SpotIt system [5]) make better composites than images synthesized directly from actual photographs (as used in our system and the CAFIIR system [31]). This is essentially an issue of representation. Which representation is better for composites, Eigenfeature coefficients or a set of indices into a database of face parts? (This latter representation is what we used, i.e., our composite faces can be reconstructed from a list naming the database faces whose parts they use.). An advantage to the Eigenfeature coefficients is that one can use the same representation both for searching the database and for creating the composite, so the system doesn't need to make a translation from one representation to the other, e.g., before conducting a search on a composite. But this advantage is not very important as long as any translation is fast and easy. In any case, both representations must be translated into a bitmap image for displaying the composite face to the user. Translation from this bitmap image to Eigenface coefficients is fast and easy (just a series of vector multiplications and additions). Even if one is using an Eigenfeature-based representation for search, as long as the system keeps track of the feature locations in the constructed composite (and this is easy to do if the composites are constructed from database images with feature-location annotations), translating the composite into its Eigenfeature representation should still be quick enough for interactive use. In this case the vector multiplications and additions are done for each feature-based sub-image (Actually, this need be done only for painted composites since unpainted composites inherit their feature-based coefficients directly from their "parents"). Thus the speed of translation is probably not a big issue, at least for these two representations. Still, from an aesthetic standpoint the idea of a single unified

representation for both database search and composite creation is appealing. This is especially true because composite creation and database search appear to be such related tasks. Creation is a search in the "canvas" space of possible faces, while database search is a more limited search in the space of the database faces.

The primary motivation for using two different representations for database search and composite creation is simply that one wants to use the best representation for the task at hand, and the representation used for search may not be the ideal one for image creation. The representation that is used not only defines the size of the space, but also which other images are the closest neighbors of an image in the space (i.e., the neighborhood), both of which are important considerations. If an image is represented by eigenface or eigenfeature coefficients, then its neighbors are other images with similar coefficients. If the image is represented by a set of indices into a database of face-parts, then its neighbors are faces with only one (or maybe two) of those parts changed. The space of Eigenfeature coefficients is more smooth and continuous than the space of database face-part indices, but it is also potentially much larger. The size of the space of *Eigenfeature coefficients* is equal to the number of possible coefficient values raised to the power of the number of feature vectors. The size of the space of *indices into a database of face-parts* is equal to the number of faces in the database raised to the power of the number of feature parts. The number of feature parts (e.g., noses, mouths, and eyes) is certain to be the smaller exponent. From the composite creation standpoint, the bigger space of eigenfeature coefficients does not appear to be advantageous, since the "smaller" space of face-part indices is amply large to express a good likeness to almost any human face. Our experiments creating composites out of 100 random faces (see Chapter 4, Section 4.4)

indicate that this would be true even with a relatively small database. The bigger space of Eigenfeature coefficients may actually make it more difficult to find a desired face in the space because one has so many more possibilities to sift through.

Exploring the neighborhood of eigenfeature coefficients may also be a less intuitive and natural way of exploring changes in the appearance of a face than exploring the neighborhood of feature part indices. In the SpotIt system, which uses Eigenfeature coefficients to represent the composite, the user manipulates numerous sliders controlling the value of the coefficients. Changes in individual coefficients do not necessarily correspond to changes in facial appearance that are naturally intuitive for people. SpotIt does allow the user to manipulate whole features, e.g., by incorporating the nose coefficients into the composite. However, refinements to the nose itself are made via sliders that manipulate its eigenfeature coefficients. It is not clear which approach is more intuitive for people, or whether the added interface complexity of the sliders is worth the extra flexibility.

Another potential disadvantage to using reconstructions as composites is that the reconstructed images are, by definition, missing information, and thus tend to look blurry and vague. It is unclear to what extent this blurriness might be disruptive to the human recognition or visual system.

Ideally, a composite creation representation should:

- Be quickly translatable into whatever search representation is being used (if it is different from the search representation).

- Enable quick and easy construction of a photographic quality image of the face for display to the user.
- Span the space of possible human faces, but not represent a bigger space than is necessary in order to do this.
- Have “neighborhoods” that are intuitively meaningful to people and that can be explored easily via some interface that is natural to use.

We considered using Eigen*face* coefficients for composite creation, and rejected this representation on the basis of these requirements. Although it easily meets the first requirement because it needs no translation, we felt it was deficient in all the others. SpotIt’s hierarchical representation, in which the face is represented both by whole features, and within a feature by its Eigenfeature coefficients, is an intriguing idea. But it is not clear how well it stacks up with respect to the above requirements. An argument can be made for either representation, but only a user study that is focused directly on the issue of representation (for the composite) will settle it.

6.6 Real-Time Feedback

Another issue to consider in designing a mug-shot search interface is that of feedback to the user about the effectiveness of queries. In the SpotIt system [5] and in the general database search system by Jacobs et al [15], the search mechanism is running in the background while the user is constructing the query. In a separate section of the screen, the images “closest” to the query are displayed and this display is updated continuously as modifications are made to the query image. Although there is a potential screen real-estate problem with this approach, the general idea is intriguing. Ideally, one could just work on

the composite until the target appears on the screen. In our 58 trials, there were 7 instances in which the score of an early interim composite was under 30, but the subject continued to tinker with it well beyond that point, sometimes making it much worse. Certainly in these cases, if the 30 closest images had been displayed, the user would presumably have been able to stop as soon as the target appeared. We noted in the final study that people often fussed with their composites well beyond the point where it was useful. Figure 40 shows a graph of the final edited composite scores from our study in best to worst order, compared to a graph of the best interim composites also in best to worst order (the best interim is the version of the composite with the lowest score among all the subject's interim versions). Clearly, if people had known when to quit tinkering with the composite,

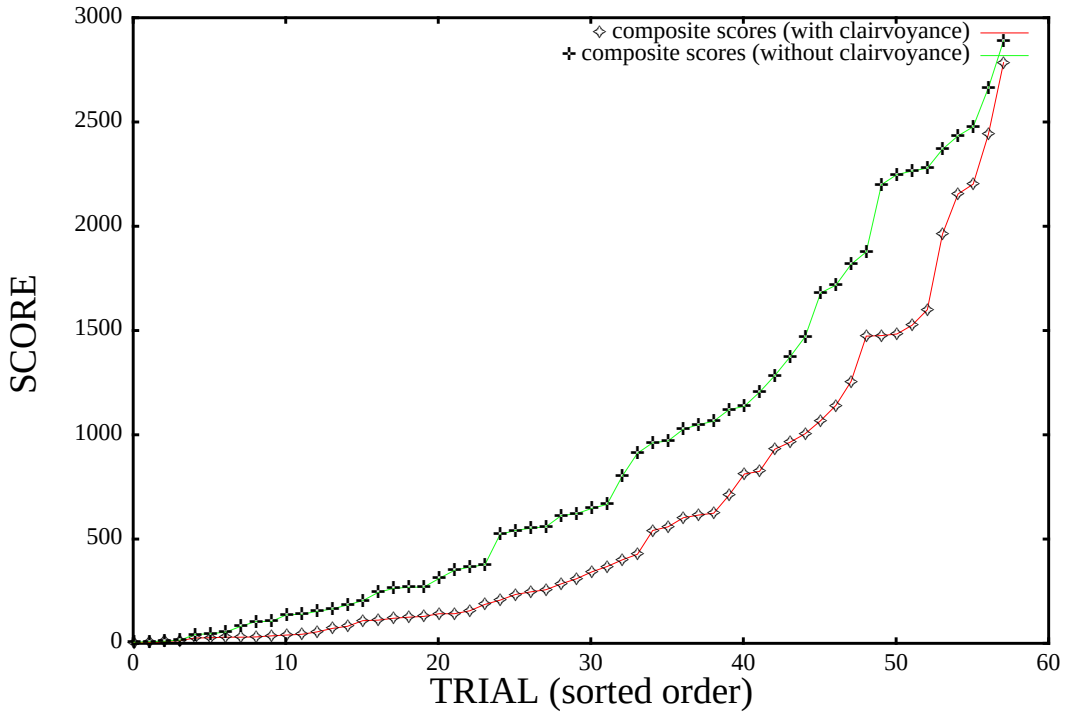

FIGURE 40. Composites: Knowing When to Stop. The upper graph shows the scores of the final edited composites, sorted in best (at left) to worst order. The lower graph shows the scores of the best interim composites, also in best to worst order. Clearly, if users had known when to stop editing, they would have created more successful composites from the perspective of "closeness" in Eigenface space. Perhaps an interface that supplies real-time feedback on the interim stages of the query would help.

they would have ended up with better scores. Real-time feedback about the effectiveness of the query might help more generally to “train” people to understand what features are most important to the search engine. This understanding may help people learn to better “communicate” their mental image to the system. Additionally, the composite may get better faster simply because features from the database faces found in interim searches can be used to improve it. In effect, this type of interface is the ultimate in integration of search and creation. Whether or not such total integration is an effective tool is essentially a search strategy issue. The potential drawback to real-time feedback is that it may greatly increase the number of image inspections. The user is constantly being bombarded with new sets of images to look over, and this may be distracting and overwhelming while simultaneously trying to work on the composite. We have already seen that hill-climbing (in which there is continual feedback about the effectiveness of queries) can fail because people do not realize when they are “close” to the target, due to problems with the similarity metric itself. Jacobs et al [15] report results indicating that a real-time feedback interface is useful for general images, but it is not clear whether their conclusion would also apply to faces.

6.7 Keeping Track of Where You’ve Been

The user may, in the course of constructing a composite, explore many regions of the database. It is useful in the course of these explorations to be able to keep track of where one has been and what one has done. It is also helpful to be able to stockpile a bunch of images (as potential queries, composite components, or “parent” images) and to look them

over together while deciding exactly how to use them. Several interface features are possible to help users avoid unnecessary travels in areas already covered and to keep track of useful images that have already been found.

In Chapter 5 we discussed the No-Review option that filters query results so database images once seen are never displayed again. During the hill-climbing portion of the study, subjects were often frustrated by seeing the same images over and over again as they tried out different but “close” query images. One subject commented that she felt like she was wandering the same hallways of a building over and over again, but never making any progress toward her destination. (This sounds a lot like the problems with local maxima that we encountered in the hill-climbing simulations.) It was clear from the simulations that the No-Review option has the potential to improve search scores and to cut down on this type of frustration. Unfortunately, it also has several serious drawbacks. One is that the target face, once missed, will never be seen again. The other is that it confuses feedback to the user about the effectiveness of a query because query results differ depending on what regions of the database have already been explored and eliminated from further consideration. Rather than not displaying images a second time, the system could simply mark previously viewed faces in some way, perhaps with a red border. Such an interface gives the user the option to disregard faces that have already been seen and focus attention instead on new faces. It may also be helpful to mark faces previously used as queries with yet a different color. As noted in the discussion on hill-climbing in Chapter 5, it was not uncommon for people to retry the same query face several times without

realizing that they were doing so. Providing a quick means of identifying those faces that have been previously seen or previously used as a query may help people avoid unnecessary cycles and to escape being trapped on local maxima.

We also added a View menu to our system that enables users to switch between several separate views of the data. Although it might be nice to make all these views visible simultaneously, issues of screen real-estate and overwhelming sensory input make this both unappealing and impractical. We are not certain exactly what set of "views" is important to maintain, but our system includes the following three. The first view is simply one's current "location" in the database, i.e., wherever one left off during the most recent query. The second view is of all faces currently saved as "parent" images for constructing random composites. The third view is a history of all images that have been manipulated by the user in any way (e.g., saved, used as a query, used for features in a composite, etc.). We chose to make the full set of random composites part of the permanent view, but these could alternatively be treated as a separate hideable view instead. Each view consists of a list of faces through which the user can page back and forth. The various views provide a means of keeping track of one's travels in both the space of the composites and the space of the database.

6.8 Summary and Discussion

In this chapter we have explored various user-interface issues and possibilities that arise when database search and composite creation systems are integrated. Random composites, a painting tool, and full-face or feature-based retrievals comprise a varied kit of interface equipment that may help to improve a user's ability to construct a good

composite. However, there is a tricky trade-off to be made between offering so many tools that the interface becomes too complex or confusing and taking full advantage of the many interface options that become possible with the integration of the search engine.

The composite representation is critical to success and we have outlined several key requirements for those considering alternative representations. Although a single unified representation for both generating the composites and searching the database sounds appealing, it may be wiser to choose two different representations if the search representation does not support the way people think about faces.

The query interface and the search strategy are closely connected. The interface determines not only what strategies are possible, but can actually impose or restrict certain strategies. An interface that provides real-time search results on interim composites is in principle the ultimate in integrating search and creation, but further study is necessary to determine whether this is actually the most effective strategy for faces. Since the mental image of a face can degrade as more and more faces are viewed, this approach may turn out to be counterproductive.

Chapter 7
Conclusions

7.1 Final Summary

Evidence that human beings process faces in a specialized way suggests that the mug-shot search problem is unique. Solutions to it draw on related work in general image database retrieval, automated face-recognition, and computer-based composite face creation. Though a few novel integrated systems have recently been reported, little work has yet been done to test these systems on real users and large datasets. Using a large (4500 image) face database and an integrated set of research software tools, we have gathered and analyzed user data to provide answers to several fundamental questions that arise in the design of a mug-shot search system. Following is a summary of the specific questions we asked and, in each case, a synopsis of our conclusions.

- *In practical "database search" terms, how effective is the Eigenface metric at similarity retrieval (i.e., at simulating human perception of similarity between faces) and precisely what is meant by this requirement?*

  When individuals selected from among 100 random database faces the five perceived to be most similar to a target face, their selections intersected with the set of closest faces to the target in Eigenface space (out of the 100) two to three times more often than would be expected if the Eigenfaces "choices" were random. In fact, in 21% of trials, people included the very top Eigenface "choice" among their five choices, whereas we would expect this figure to be only 5% if the Eigenface top choice were a random pick out of the 100. Thus a correlation between the human choices and the

Eigenface “choices” is evident. However, this correlation is not high enough to dramatically reduce the number of image inspections that would be required to find the target if people used only their single top choice face out of the 100 as a query. One fundamental problem is that there is significant disagreement among people themselves about which faces are most similar to a target, implying an upper bound on how much one can expect of Eigenfaces or any other metric. One way to define the “human” similarity metric is “the consensus of most popular choices among a group of people.” Defined this way, in our experiments, the Eigenface metric did as well or better than 29% of the study subjects at making choices that captured the consensus of a group. Since the correlation between the Eigenface metric and the “human” one, although it exists, is not overwhelming, the choice of search strategy must be made carefully so as to use this correlation to the best advantage.

- *Given an ideal computer-based similarity metric, i.e., one that truly imitates similarity criteria of a human user, how could it be used and what reduction could it achieve in the number of images the user would have to inspect before finding a target image? What kinds of search strategies are most effective with an actual mug-shot search system, and how does the performance of the similarity metric affect the choice of search strategy?*

  Using computer simulations, we showed that a search strategy of simple hill-climbing in the space of database images works extremely well in the case of an ideal metric, resulting in a search score on the order of 2% of the database. This was better than for any other strategy we simulated. However, the excellent performance of hill-climbing in principle is not sustained in practice, given the actual level of correlation between

the Eigenface metric and the “human” one. The study found that the average number of image inspections required for a hill-climbing search using Eigenfaces was, in fact, closer to 35% of the database. This is an improvement over the 50% required, on average, for a simple sequential search of the data, but it is not good enough to be practical for large databases. On the other hand, a non-iterative strategy of constructing a single composite query image from a set of 100 random database faces appeared to be superior to hill-climbing. In the study, this simple composite strategy reduced the average number of image inspections to about 20% of the database, a number that begins to enter the realm of a practical system. In general, the use of facial composites (as opposed to restricting users to database images for their queries) proved to be beneficial.

- *How might one exploit the integration of systems for database search and composite creation to produce a better query interface? What are the critical components in the design of the interface and what are the advantages, disadvantages, and trade-offs that must be considered?*

  There is a critical trade-off in any interface between simplicity and added functionality. The integration of search and creation makes possible many flexible new ways to construct and manipulate a query (i.e., composite), but care must be taken to ensure that each new added function is truly useful and does not unnecessarily complicate the interface. Possible interface ideas that we and others have explored are random composites, feature-based retrieval, painting on composites, maintaining multiple views, and simultaneous feedback (i.e., real-time search results in response to every modification of the composite). The underlying

representation used for composites may be the same as or different from that used for search, but whatever composite representation is used must support the way people think about faces. With the integration of search and creation, the composite construction interface and the search strategy become closely connected. The interface can determine not only what strategies are possible, but can actually impose or restrict certain strategies. Our study showed that composites make more effective queries, but the degree to which search and creation should be intertwined is still unclear. Further study is needed to determine exactly what features (and hence what strategies) the interface should support.

7.2 Future Work

There are three main avenues for seeking improvement in mug-shot search systems. The first is to attempt to improve the correlation between the human and system metrics for determining similarity between faces. The second is to determine search strategies that best exploit whatever correlation does exist and attempt to build those strategies directly into the system. The third is to seek a query formulation interface that best facilitates easy construction or location of a query image matching the mental one. There is potential for improvement in each area, and progress in one area may affect progress (or the need for it) in another. The performance of the similarity metric affects the choice of search strategy, which, in turn, is closely tied to the requirements of the query interface.

Many face-recognition techniques have been studied for their ability to identify faces, but little has been done to determine their effectiveness at the very different task of assessing perceptual similarity between faces. Our study used a simple version of the

Eigenface similarity metric, but many parameters such as the size of the training set, the particular eigenvectors to use, and the specific distance calculation (e.g., Euclidean, Mahalanobis, etc.) can affect the performance of the metric. The literature is laden with variations on the Eigenface metric (e.g., [19], [23], [30]) as well as with many different metrics (e.g., [15], [18], [21], [27]), but we do not know which of these many possibilities is best for mimicking human perception of facial similarity. Competitions have been held to determine the face-identification ability of face-recognition metrics [26]. Using our evaluation method (e.g., search scores) for assessing ability at similarity retrieval, a similar competition could be conducted to compare the performance of candidate similarity metrics. Only a single user study would be necessary, but the data (i.e., the database, together with the selections made by study subjects) could be analyzed with each candidate metric in turn to see which one produces the best average search scores. This would provide a fair and uniform comparison of the metrics and would ensure that each candidate metric is implemented as its designers/proponents intend.

Our study showed that the random-set strategy using a composite was more effective than hill-climbing without composites. Further study is needed to determine whether hill-climbing with composites (i.e., in which the composite is iteratively improved upon with intermediate search results) is even better, and whether one should use the same or different underlying representations for the search and creation subsystems. In general, the degree to which the query interface and the search capability should be intertwined is still unclear. At one end of the spectrum is a simple concatenation of the two systems for search and creation, in which a complete composite is created first and then submitted to the search engine. At the other end of the spectrum is total integration, in which every

single modification to the composite produces search results that may be used to update the composite. Further user studies specifically focused on varying approaches to the query interface are needed to determine which strategies along this continuum of integrated strategies are best for searching a database of faces.

Appendix A
Pilot and Final Study Results

The two tables below contain raw study data from the Pilot and Final studies. There is one row for each trial showing the image scores for the subject's top five database choices (out of the 100 random faces), given in the subject's rank ordering, most similar to least similar (left to right). These are followed by scores for the subject's random composite selection and edited composite.The lowest score for each subject is shown in boldface. The scores shown are those that would be obtained had that image been used as a query.

TABLE 5. Pilot Study Results.

Target	database 1	database 2	database 3	database 4	database 5	composite random	composite edited
1	137	1013	1230	206	1510	133	**11**
	1230	40	2333	993	137	24	**1**
	1230	993	1736	**40**	206	401	272
	137	237	1602	40	2333	7	**0**
	1230	40	137	1510	206	**7**	10
	1230	1602	437	40	1013	**2**	11
	942	1602	**137**	1736	1230	206	825
	942	**137**	237	1602	713	451	879
	1230	**40**	1602	993	2542	216	1831
	40	2333	1230	942	1736	862	665
	40	1873	206	942	1230	744	496
2	**39**	2058	2635	2218	1333	2036	1967
	2348	1907	293	1333	1619	458	**88**
	2218	1619	96	1333	367	1926	**42**
	2218	970	1534	788	225	1060	**53**
	808	**39**	293	2218	262	72	144
	293	878	2058	223	**49**	152	140
	808	536	**39**	2348	878	1323	608
	1333	921	2635	143	**49**	1376	935
	536	2218	2081	293	396	**208**	223
	536	808	**396**	2370	1111	724	1067
	2483	**1496**	2218	2635	2535	1995	2580

TABLE 6. Final Study Results. This table shows image scores of the top five database images (out of the 100 random faces), the random composite, and the edited composite for each trial in the final study. The lowest score for each is shown in boldface.

Target	database 1	database 2	database 3	database 4	database 5	composite random	composite edited
A	1350	305	2257	**108**	2254	2132	649
	305	351	2257	1350	3268	**81**	84
	2884	1350	3268	2891	**358**	1096	525
	1350	**305**	3268	3184	3373	1879	1375
	2257	1350	2979	524	**157**	483	270
	524	1350	**108**	1289	157	666	166
	2257	1350	3172	1875	**358**	2013	1286
	1350	967	2257	166	**108**	326	313
B	2161	311	2738	1374	185	**31**	542
	185	722	311	1082	3565	344	**47**
	2237	2861	**1082**	1374	2234	2137	2479
	2433	2161	3054	2738	2449	2498	**1122**
	2539	2161	**365**	2433	2024	1280	1819
	185	2503	1046	1723	2433	1499	379
	185	2161	2024	2738	3730	1875	**12**
	2503	2449	**2161**	3282	3344	3670	2666
C	3038	2732	2007	3378	2203	3240	**1472**
	2575	2249	236	1424	2711	1667	**106**
	1424	2428	2732	**236**	1875	1988	554
	1875	2007	2428	**731**	1199	1492	1682
	1362	3038	2732	3781	2931	2381	2372
	781	1655	254	596	**214**	1285	246
	1850	3378	3038	3407	2007	3165	2892
	3038	**236**	1850	3378	3781	544	914
D	**318**	2490	3295	1242	673	1328	1139
	318	1730	397	180	673	**125**	265
	930	212	269	673	1532	**83**	1029
	2490	2485	868	**180**	1118	2014	1879
	397	930	**98**	1034	748	115	352
	493	135	314	1768	1118	221	**44**
	1034	2147	673	318	**180**	324	614
	2490	602	1118	2187	1034	669	**367**
	673	**318**	3146	2490	3295	1444	1206

TABLE 6. Final Study Results. This table shows image scores of the top five database images (out of the 100 random faces), the random composite, and the edited composite for each trial in the final study. The lowest score for each is shown in boldface.

Target	database 1	database 2	database 3	database 4	database 5	composite random	composite edited
E	256	1094	195	1958	2983	2916	**9**
	1094	2103	2254	3746	2428	3792	2201
	256	3057	3746	910	2983	590	**141**
	256	2983	3746	3158	1744	263	**155**
	256	2103	3158	3105	2590	2871	972
	3746	256	2254	481	220	155	**7**
	1952	3158	3107	256	**220**	2971	963
	481	220	910	3374	3105	728	**138**
F	1461	1636	1528	**31**	261	1198	623
	693	261	1636	**16**	2967	635	206
	1528	1829	1636	270	613	1027	**271**
	1636	1660	2747	2967	2913	2693	**668**
	270	1461	1275	483	736	639	561
	500	1305	1275	270	1636	110	**107**
	693	**16**	261	1746	1461	474	17
	270	1275	1528	**31**	2855	1184	55
	613	483	261	**31**	1461	739	803
G	205	2531	**56**	892	1191	709	2281
	1900	2241	259	**56**	3109	1733	1719
	2064	892	2531	1845	**302**	2195	1067
	1191	1900	2322	2241	1688	1831	**1050**
	1620	1688	2739	1206	**425**	1951	2267
	136	1845	302	**56**	2743	357	186
	3068	2739	**425**	2064	2743	2864	2250
	1292	939	842	**302**	2521	2170	2436

Appendix B
Proof of (D-N+1)/(N+1) Analysis

In Section 5.5 we state that, based on a simplifying assumption, the expected score of the best of N random selections from a database of size (D+1) (i.e., a database consisting of elements labelled 0,1,2,...,D) is (D-N+1)/(N+1). In this appendix, we explain our assumption and provide a proof.

Definition: The *score* of image P_1, with respect to image P_2, is the position (or rank) of P_2 in the list of images obtained by sorting the database images by distance from P_1. (Note: If P_2 is the target, then the score of P_1 corresponds to the number of image inspections required to find the target if P_1 is used as a query.)

Definition: The *best* out of N images selected at random from the database is defined as the image with the lowest score (with respect to a target, T).

Simplifying Assumption: *The score of P_1 with respect to P_2 is equal to the score of P_2 with respect to P_1.* This is synonymous with saying the following: If, in the sort on P_1, there are x images closer to P_1 than P_2 is to P_1, then, in the sort on P_2, there are also x images closer to P_2 than P_1 is to P_2 (note that they are not, in general, the same x images). Given a large enough database, a uniform distribution in the space, and barring issues that come up at the "edges" of the volume defined by the database, for the purposes of our analysis this is a reasonable assumption.

Lemma: If N numbers are selected at random from the set of integers 0, 1, 2,...,D, then the expected value of the lowest valued number selected is (D-N+1)/(N+1). [4]

Claim: For a database of size (D+1), the expected score of the best of N random selections is (D-N+1)/(N+1).

Proof: Suppose we select N random numbers (n_1, n_2, n_3, ..., n_N) between 0 and D, inclusive. Consider the list obtained by sorting the database by distance from target T. In fact, one does not know the target image a priori, but one can imagine such a sorted list in principle. The set of images whose positions in this list are n_1, n_2, n_3, etc., can then be considered a set of N random selections from the database. Let image P be the image in this set with the lowest valued position or rank, Nlow. By the Lemma, the expected value of Nlow is (D-N+1)/(N+1). This is the expected score of T with respect to P. By our simplifying assumption, this is equal to the score of P with respect to T. So the expected score of P with respect to T is also (D-N+1)/(N+1).

Appendix C
100 Random Faces

Following are the 100 faces that were chosen randomly from the database of 4500 images. These 100 faces were used for both the pilot study and the final study experiments. They are presented in five pages of 20 images, just as they were presented to subjects in the study.

References

1. E. Baker and M. Seltzer. “The Mug-Shot Search Problem.” *Technical Report TR–20–97, Harvard University Center for Research in Computing Technology,* 1997.

2. E. Baker and M. Seltzer. “The Mug-Shot Search Problem.” In *Vision Interface ‘98 Proceedings,* pages 421-430. Canadian Information Processing and Pattern Recognition Society, June 1998.

3. E. Baker and M. Seltzer. “Evolving Line Drawings.” *Proceedings Graphics Interface ‘94,* Wayne Davis and Barry Joe, Editors, pages 91–100, May 1994. Morgan Kaufmann Publishers.

4. A. Baker and E. Baker. “Expectation Value of the Lowest of a Set of Randomly Selected Integers.” *Technical Report TR–11–98, Harvard University Center for Research in Computing Technology,* 1998.

5. R. Brunelli and O. Mich. “SpotIt! an Interactive Identikit System.” *Graphical Models and Image Processing*, Vol. 58, No. 5, pages 399–404, September 1996.

6. R. Brunelli and T. Poggio. “Face Recognition: Features vs. Templates.” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 15, No. 10, pages 1042–1052, October 1993.

7. C. Caldwell and V. S. Johnston. “Tracking a Criminal Suspect Through Face Space With a Genetic Algorithm.” *Proceedings of the Fourth International Conference on Genetic Algorithms*, pages 416–421, 1991, Morgan Kaufmann Publishers.

8. C. Caldwell and V. S. Johnston. “Tracking a Criminal Suspect Through Face Space With a Genetic Algorithm.” *Handbook of Evolutionary Computation*, Oxford Press.

9. R. Chellappa, S. Sirohey, C. L. Wilson, and C. S. Barnes. “Human and Machine Recognition of Faces: A Survey.” *Proceedings of IEEE*, Vol. 83, No. 5, pages 705–740. May 1995.

10. H. D. Ellis. “Introduction to Aspects of Face Processing: Ten Questions in Need of Answers.” *Aspects of Face Processing*, pages 3–13, Dordrecht: Nijhoff, 1986.

11. M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, P. Yanker. “Query by Video and Image Content: The QBIC System.” *IEEE Computer*, Vol. 28, No. 9, pages 23–31, September 1995.

12. D. E. Goldberg. *Genetic Algorithms in Search Optimization and Machine Learning*. Addison-Wesley, 1989.

13. V. N. Guidivada and V. V. Raghavan. “Content-Based Image Retrieval Systems.” *IEEE Computer*, Vol. 29, No. 9, pages 18–22, September 1995.

14. P. J. B. Hancock, V. Bruce, and A. M. Burton. “A Comparison of Two Computer–Based Face Identification Systems With Human Perceptions of Faces.” *Vision Research*, in press, 1998.

15. Charles E. Jacobs, Adam Finkelstein, David H. Salesin. “Fast Multiresolution Image Querying.” Proceedings of SIGGAPH 95, in *Computer Graphics Proceedings*, Annual Conference Series, pages 277–286, August 1995.

16. M. Kirby and L. Sirovich. “Application of the Karhunen-Loeve procedure for the Characterization of Human Faces.” I*EEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, No. 1, 1990.

17. W. Konen. “Comparing Facial Line Drawings with Gray-Level Images: A Case Study on Phantomas.” Retrieved from the web: http: // www.zn.ruhr-uni-bochum.de

18. M. Lades, J. C. Vorbrüggen, J. Buhmann, J. Lange, C. von der Malsburg, R. P. Würtz, and W. Konen. “Distortion Invariant Object Recognition in the Dynamic Link Architecture.” *IEEE Transactions on Computers*, Vol. 42, pages 300–311, 1993.

19. A. Lanitis, C.J. Taylor, and T.F. Cootes. “An Automatic Face Identification System Using Flexible Appearance Models.” *Proceedings of the 5th British Machine Vision Conference*, Edwin Hancock, Editor, Vol. 1, pages 65–74, 1994. BMVA Press, York, UK.

20. T. Minka. “An Image Database Browser that Learns From User Interaction.” *MIT Media Laboratory Technical Report 365*.

21. P. S. Penev and J. J. Atick. “Local Feature Analysis: A General Statistical Theory for Object Representation.” *Network: Computation in Neural Systems*, Vol. 7, No. 3, pages 477–500, 1996.

22. J. Penry. Photo-Fit. *Forensic Photography*, Vol. 3, No. 7, pages 4–10. 1974.

23. A. Pentland, B. Moghaddam, and T. Starner. “View-Based and Modular Eigenspaces for Face Recognition.” *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, July 1994.

24. A. Pentland, R.W. Picard, and S. Sclaroff. “Photobook: Tools for Content–Based Manipulation of Image Databases.” *International Journal of Computer Vision*, Vol. 18, No. 3, pages 233–254, 1996.

25. Phantomas. Product Description on the web: http: //www.zn.ruhr-uni-bochum.de.

26. P. J. Phillips, H. Moon, P. Rauss, and S. A. Rizvi. “The Feret September 1996 Database and Evaluation Procedure.” *Proceedings of First International Conference on Audio and Video-Based Biometric Person Authentication*, Crans-Montana, Switzerland, March 1997.

27. S. Ravela and R. Manmatha. “Image Retrieval by Appearance.” *Proceeding of SIGIR 97*, Philadelphia, July 1997.

28. G. Rhodes, S. Brennan, and S. Carey. "Identification and Ratings of Caricatures: Implications for Mental Representations of Faces." *Cognitive Psychology* 19, pages 473–497, 1987.

29. M. Turk, A. Pentland. Eigenfaces For Recognition. *Journal of Cognitive Neuroscience*, Vol. 3, No. 1, pages 71–86, May 1991.

30. W. A. S. M. Wahid. "Toward Building a More Practical Face Recognition System." *Masters Thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology*, June 1997.

31. J.K. Wu, Y.H. Ang, P. Lam, H.H. Loh, and A. Desai Narasimhalu. "Inference and Retrieval of Facial Images." *Multimedia Systems*, Vol. 2, No. 1, pages 1–14, 1994.

32. A.L. Yuille. Deformable Templates for Face Recognition. *Journal of Cognitive Neuroscience*, Vol. 3, No. 1, pages 59–70, 1991.

33. A.L. Yuille, D.S. Cohen, and P.W. Hallinan. "Feature Extraction from Faces Using Deformable Templates." *International Journal of Computer Vision*, Vol. 8, No. 2, pages 99–111, August 1992.