2015 IEEE 2nd International Conference on Cyber Security and Cloud Computing

Face Recognition with Environment Tolerance on a Mobile Device

Philip J. Riesch, Xiaojiang Du, Haibin Ling
Department of Computer and Information Sciences
Temple University
Philadelphia, Pennsylvania
{philip.riesch, xjdu, hbling}@temple.edu

Michael J. Mayhew
Cross Domain Information Systems Group
Air Force Research Laboratory
Rome, New York
michael.mayhew.1@us.af.mil

***Abstract*—One of the most logical applications of face recognition for authentication is on mobile handset devices. However, face recognition still faces challenges in providing *environment tolerance*: being able to compensate for changes in light conditions within an environment where authentication is occurring, due to users carrying their mobile handset devices to different locations with varying and unpredictable sources of illumination. Existing face recognition systems operate by finding fiduciary points relative to the area of the entire face, which becomes their weakness when they are not used in applications where light conditions are fixed and controlled. This research investigates *Local Binary Patterns* (LBP), an image encoding technique whose origins lie in texture analysis, in order to overcome the problems faced by existing face recognition systems and provide tolerance to variable light conditions. This research aims to utilize LBP on modern mobile handset device hardware that is "off-the-shelf": utilizing only the most basic and widely available onboard imaging hardware and processing capability provided on mobile handset devices of the present day. We have performed rigorous experimentation with LBP both on large databases of images of human faces, as well as developing mobile handset software that was deployed to real users and tested in a field environment. Our experimentation indicates that LBP is capable of being used to develop face recognition systems that provide environment tolerance, potentially finding practical use as a component of mobile device authentication applications.**

Keywords—computer vision; face recognition; feature extraction; mobile computing; authentication.

I. Introduction

It's one of the most prominent clichés of the smartphone revolution: *"Oh one day you will just be able to hold your phone up to your face and it will recognize you and unlock!"* Call it the *flying car* cliché of the predicted future of mobile handset software. However, while the nature of clichés would expect us to acknowledge these predictions as nothing but futuristic pipe dreaming, instead face recognition technology is something that we have come very close to achieving in a practical-enough context that we can squeeze it into practical applications and make it work for us. But, one major obstacle still stands in our way – compensation for variable environment conditions: bright light or dim light, inside or outside. The purpose of this research is not only to discover face recognition systems that are *environment tolerant*, but also to make such a system compact enough to operate on mobile handset devices in a practical application.

The notion of enviornment tolerant computerized face recognition is a task that transcends across a broad landscape of scientific and mathematic discipline. To fully understand face recognition, we must understand the lineage of past research efforts that has guided our own research to come to the conclusions that it has. We need to understand why the most logical-seeming approaches have historically been the least effective. To come to understanding required us to take a very broad survey across the topic's entire research history.

After we discover a face recognition system that is suitable for fulfilling our objective of environment tolerance, we are then faced with the additional challenge of getting such a solution to operate on a mobile handset device. We must look at the problem from several facets: on one hand, we need a face recognition system that is tolerant to changes in the environment: mobile handset devices, for instance, are carried by their users to all sorts of places with all sorts of light sources that have different properties. Face recognition that is environment tolerant is a challenge that has traditionally been difficult for face recognition systems to overcome. The challenge is then amplified when whatever solution we develop then needs to be constructed to be efficient enough to be able to operate reliably on the limited resources of a mobile device.

The challenges of our research, therefore, are twofold. We must survey face recognition technologies in order to discover approaches to face recognition that are theorized to be tolerant to environment variability per our definition. We must test and verify that these methods are acceptably accurate enough that they may be used for the purpose of authentication. And, we must find a way to make the method that we choose nimble enough that we can run it in a standalone configuration on a mobile handset device. If these challenges can be overcome, then perhaps mobile phone biometric authentication will finally be able to have its *flying car* moment.

II. A Review of Contemporary Face Recognition Literature

Electronic face recognition is a problem that has been under scrutiny by researchers and mathematicians for more than 50 years, and to this day we still have not been able to develop a solution that works reliably under any given set of conditions.

978-1-4673-9300-3/15 $31.00 © 2015 IEEE
DOI 10.1109/CSCloud.2015.79

IEEE computer society

We need to be able to compensate for the environment – light conditions, specifically – because if the current weather or what kind of light fixtures you have in your office can throw off or disable your recognition system, then how can we ever be able to trust it at all? It is enough of a challenge to develop face recognition systems that are accurate and reliable under *sterile* light – static light sources that never change in strength, orientation, or wavelength – so to then find a way to compensate for variable light creates additional challenges.

A. "The Lay Man's Solution": Geographic Feature Mapping

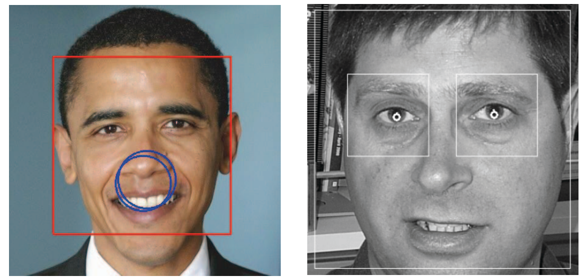
If one were to ask a data scientist that is not familiar with face recognition – or any person really, computer scientist or not – how to do face recognition, you may find that their answer is somewhat predictable. They will logically conclude that the solution to the problem is focusing on the details of a face that we as humans *believe* we are analyzing when we personally try to identify faces ourselves. When a human is trying to identify another person, we may focus our attention on clearly defined geographic features such as the eyes: how big they are, what color they are, the position and size of the nose and lips, or we may focus on other distinguishing features such as eyebrow size and shape or hair color or some other clearly interpreted detail about someone's face.

But this approach becomes very complicated very quickly. Sure, we as humans can identify someone's face: we can identify their eyes and we can identify their nose and we can identify where those features are relative to other points, but we as mammals have a complicated visual cortex that has evolved over millions of years to deduce where these abstractions exist from within the overwhelming quantity of raw visual information that is streaming into our heads at all hours of the day.

The exact specific biological mechanism as to how this all works is something that is itself under debate by scientists. So then, we are presented with a serious problem: we are trying to create a computer emulation of a human biological mechanism which we don't know how it really even works in the first place, and on top of that because it is a computer we expect it to be more accurate than a human, ideally. Frankly, it's a bit like trying to build a nuclear reactor while having only a nineteenth century understanding of nuclear physics!

This does not necessarily mean that it is not possible, however. Some of the earliest examples of face recognition research have been guided by the notion that it is possible to digitally replicate what we perceive as being the human way of performing face recognition: "*the lay man's solution*." The earliest examples studied by our research are systems based on **Hough Circle Transforms (HCT)**. These early face recognition systems were built on a hypothesis that a series of equations that were originally developed for electronically processing the bubble chamber photographs generated by 1950s-era nuclear physics research could also be used to recognize and identify human faces.

An early system developed by a team in Southampton, England in 1985 epitomizes this theory [1]. They postulated that performing a Sobel edge detection [2] on a grayscale picture of a face, then using HCT [3] to mark and measure the distance between the eyes and the eyebrows could be used to encode and determine identity. In practice, however, our research found that this system is far too simple to provide any sort of consistency in detecting the landmarks that we are interested in; oftentimes this system was much more wrong than right (Figure 1).



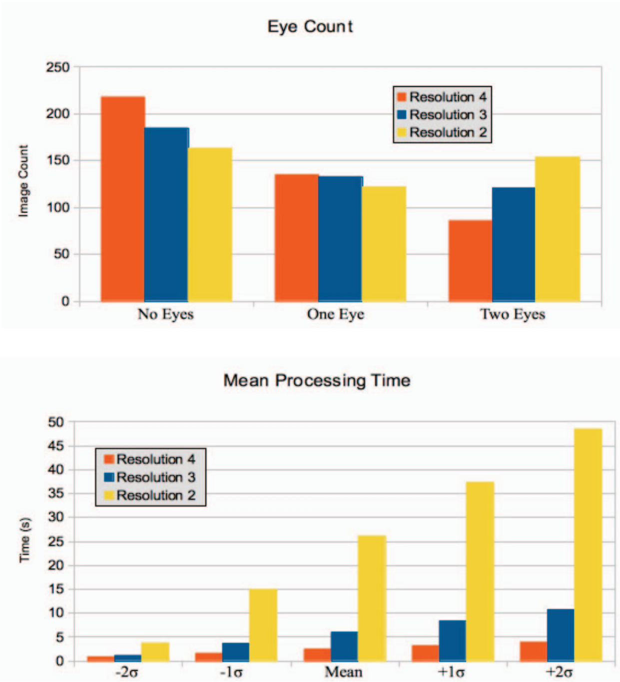
Figures 1 & 2: (Left) HCT on a human face. This image was processed with a Canny edge detector [4] rather than a Sobel edge detector. HCT was an inefficient method for performing eye localization. (Right) Iris localization by means of Timm-Barth method [5] of gradient orientation analysis.

But at the time, our research was still in an initial naïve phase of its understanding, and so we therefore searched for ways of obtaining more accurate detection and marking of prominent landmarks on the face. We had not yet proven nor disproven that creating a map of measurements from these points could effectively encode identity, since our initial approach to discovering the location of these points – influenced by the work of the Southampton team – was too simple, and too inaccurate. Our research therefore focused on studying more modern eye localization algorithms … perhaps more accurate iris localization techniques would enable us to obtain reliable mappings of the locations of biological structures on the face. A method for localizing the position of eyes in a two-dimension image of a face that was captured with basic RGB imaging hardware, originally developed by a German team in 2011 for use in gaze tracking [5], was studied in order to determine whether more accurate mapping of prominent biological features was even possible (Figure 2).

At it's best performance, our implementation of this eye localization algorithm was able to correctly identify the position of both eyes in a picture of a face only about 30% of the time, and at least one eye about 60% of the time (Figure 3). And unfortunately, these rather lackluster results came at a massive time and resource cost (Figure 4). We therefore were able to conclude that the reason geographic feature mapping is ineffective is not because we are unable to adequately encode a person's identity from the measured distance between the biological structures of the face, but rather because finding those structures is itself something that is an inherently difficult challenge to overcome. A whole lot of processing power is used to achieve relatively mediocre results.

B. Achieving Environment Tolerance through Refined Resolution: Local Binary Patterns

Realistically, it cannot always be guaranteed that the environment where authentication is performed is going to be constant. This is especially true if one were interested in developing a mobile authentication platform: such is the case with the ultimate objectives of this research effort.



Figures 3 & 4: Eye localization implementation and testing. In the context of face recognition, this system was found to be inadequate in both it's accuracy (above) and it's utilization of computing resources (below).

So therefore, the next technological evolution of face recognition is developing new statistical methods that are capable of compensating for unpredictable light scenarios; providing resonably accurate identification performance. Our research began to study technologies that have been derived from the study of computerized texture analysis.

One such texture analysis system that shows promise for use in environment tolerant face recognition is a texture analysis technique called **Local Binary Patterns (LBP)**. In LBP one generates a *texture map* from an input image by comparing each pixel in an image to a neighborhood of pixels that immediately surround that pixel [6]. "Classic" LBP compares the pixels that are directly adjacent to the target pixel. Newer LBP schemes, called "extended" LBP, have also been developed that compares each pixel to a neighborhood whose sample size and distance from the target pixel are variable parameters [7]. Comparison with each neighbor pixel is a threshold operation that generates one bit of texture data for each pixel in the source image, totaling eight bits altogether for all pixels in the neighborhood. These bits of data combine to yield an *LBP Descriptor* for a given pixel, and these LBP Descriptors combine to yield the texture map (Figure 5).

Figure 5: Application of LBP: before (left) and after (right)

The original intention of LBP was as a texture analysis tool: identifying specific local combinations of LBP descriptors in the texture map could theoretically be used to analytically identify and label lines, holes, edges, and other specific shapes that might exist in the input image [8]. In the context of face recognition research, however, we are not interested in identifying fiduciary points from textures. Instead, our methodology aims to take much more of a holistic approach towards encoding user identity.

III. Face Recognition Using Local Binary Patterns with Support Vector Machines

Some of the earliest systems that utilize LBP in order to perform face recognition begin to appear around 2004. A Finnish team first acknowledged the problems faced by face recognition systems when they are not used in controlled environments [9]. This paper lays an important piece of groundwork for our system: when LBP is used in texture analysis, we are normally concerned with finding specific regional patterns of descriptors in the texture map. But, in face recognition we wish to distill the texture map into a single data structure that acts as a composite representation of the entire face. So therefore, once we have generated our texture map, we then want to generate a *histogram* of the texture map. This histogram in essence becomes a pseudo-fingerprint of the person's face, and we use this histogram in order to encode and store identity.

Histograms are generated using a regional methodology. We divide the texture map into a grid of regions. Each of these regions then has a histogram calculated for that region. Once we have generated histograms for each of the regions, these regional histograms are then concatenated together in order to yield a composite histogram of the texture map. This composite histogram becomes the ultimate data structure that we use to encode and store identity during enrollment, and later on to evaluate identity during authentication.

When using LBP to perform face recognition, generating the composite histogram of the texture map is generally a common precursor step for all systems that use LBP as the basis of their system. However, what happens after the histogram is generated is where the various LBP systems begin to diverge from each other. The common goal that is trying to be achieved is to feed the histogram into a statistical model that enables the system to make a comparison of the histogram from an unidentified individual against a database of known people, whose histograms – labeled with their identity – have been fed into the statistical model as training data during it's enrollment phase. Early LBP face recognition systems tended to use nearest-neighbor classifiers such as *k-nearest, log-likelihood,* and *chi* square [9].

However, our research feels that these very simple statistics models – while suitable for texture analysis – are not suitable for general-purpose face recognition. Our research focused on the application of **Support Vector Machines (SVM)** for use in this role. One of the works studied by our research that involved using LBP in combination with SVM to perform face recognition is a 2008 work by a team based in Israel who were interested in investigating *pair matching* [7]. Their work in evaluating LBP with SVM for face pair matching entailed using a labeled image database of subjects [10] processed with LBP, in which a subset of histograms labeled as being a particular person were used with all histograms in the database that were labeled as not being a particular person as the training data for the SVM model. This technique, called *One Versus All*, used a very large sample of negative vectors and a rather small sample of positive vectors in order to train the SVM. A subset of histograms labeled as being the person whom the model was trained for, but were not already used to train the model, were used to test and verify the accuracy of the trained SVM model.

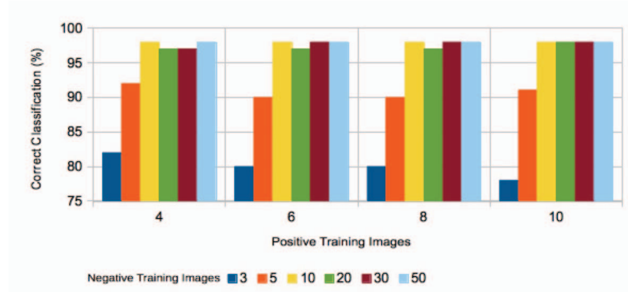
A. Evaluation of LBP with SVM

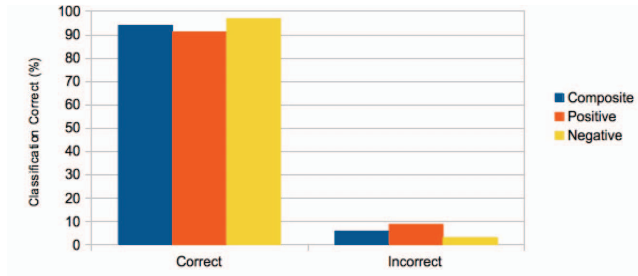
The method we designed to begin an evaluation of this face recognition system was an implementation of "standard" LBP that distilled texture maps into a composite histogram from square-shaped subregions evenly distributed across the texture map. These histograms were then fed into an SVM for classification. Our initial prototype was developed in Python, using the OpenCV computer vision library [11] to provide image manipulation functions and machine learning models.

Deviating from other face recognition research that has been done in the past, we elected to not use the same face databases that other research efforts have utilized, such as the FERET database [12] and the Labeled Faces in the Wild (LFW) database [10]. For our research, we chose to use the much lesser known Caltech Front Face database [13], and we made this choice for several reasons. Firstly, the images in the Caltech database are very high resolution relative to other well known databases, about the resolution that could be generated with modern mobile phone camera hardware. Second, the database contains at least fifteen to twenty images of one subject for approximately ten of the 30 subjects in the database, allowing us to build and test many different combinations of training sets and testing sets for the same labelled subject. And thirdly, we chose this database because it contains ample images of individuals in a variety of environment and light conditions, from dim nighttime settings to bright indoor settings to very bright outdoor settings.

One of the first things we sought to determine is to discover what is the minimum set of training images (positive labeled and negative labeled) that are needed in order to obtain an acceptable degree of recognition accuracy. We understand from reviewing prior research that the positive training image set and the negative training image set can be asymmetric: they do not necessarily need to be equal in size. So what is the optimized, while still adequately minimal, count of how many positive and negative histograms we should feed this system during it's training phase?

To find the answer to this question, we performed a series of experiments. We labeled the Caltech database, giving us sets of images that are labeled as being images of the same subject. From those initial labeled sets, we then created additional sets of images that were either "positive" sets: known to belong to the subject being tested, and "negative" sets: known to not belong to the subject under test. For each iteration of this initial experiment, we used positive set sizes of 4, 6, 8, and 10 images, and we used negative set sizes of 3, 5, 10, 20, 30, and 50 images. These combinations of positive sets and negative sets were combined together to build the training sets that we distilled into histograms and used to train an SVM. In total, 24 different training sets were generated per subject tested.





Figures 6 & 7: (Above) Determining minimum histogram set size. (Below) "Worst case" SVM training and evaluation.

In each round of the experiment, once the model has been trained, we then used labelled images that were not included in the training set in order to evaluate the accuracy of the system. We tested the trained SVM against five sets of positive images in order to check positive correctness (a "positive" test), and we tested against five sets of negative images in order to check negative correctness (a "negative" test). Each test set contained 10 images. Counting all positive test images and negative test images with each permutation of training sets, we performed in total 2400 individual image comparisons. Our goal was to find the minimum effective set size of training images.

Our expermentation found that the variable that appeared to have the largest influence on the accuracy of identification was the number of negative images that were included in the training set during the SVM training phase. We found that it was possible to approach a correct identification accuracy of almost 98% when the SVM was trained with a training set that contained as few as 4 positive samples of the subject and 10 negative samples (Figure 6). However, we were skeptical about the results that we received from this preliminary experiment. 98% seems a little too good to be true. We deduced that such a high accuracy could potentially be attributed to what could be described as "best case" bias: we had maybe subconsciously selected images for the training and testing sets that were ideal and contained relatively uniform light conditions.

So for the next round of experiments, we decided to try and verify our results by creating what we felt were some "worst case" scenarios in order to see if the results that we had obtained during our first series of experiments were actually realistic. The results from our initial experiment have led us to believe thusfar that the optimum minimum training set size is 4 positive samples and 10 negative samples. The Caltech database contains approximately 30 individual subjects, and 19 of these subjects have image sets that contain at least 19 images or more. The next round of experiments that we had conducted included 10 separate tests per subject: 5 of these tests focused on correct positive identification and 5 tests focused on correct negative identification.

Each test was either a positive test or a negative test, and for each test, each individual subject was tested three times. For each of these test iterations, the image set that was used was rotated so the same selection of training images were not used in any subsequent test, but the total size of the testing set was always 10. Our goal in doing this is to create multiple different scenarios of each subject both in training and in testing in order to give us a very broad evaluation under many different possible conditions.

We allowed for images to be used that were in a variety of light conditions: images that were captured both in indoor environments and in outdoor environments. In total 1140 individual image comparisons were performed in this second round of experimentation. We strived to eliminate any kind of best case bias that may have existed in our first series of experiments. The results of these experiments were just as impressive as the first series of experiments. A composite accuracy of 94% was achieved under a variety of light and environment conditions while still using only four positive and ten negative images in order to train the SVM to recognize a subject's face histogram (Figure 7).

IV. The Challenges of Mobile Biometric Authentication

Our research is not only focused solely on the objective of performing face recognition that is environment tolerant, but we also aim to deploy the technology onto mobile devices in a standalone configuration. Therefore, further developing this LBP with SVM concept to fill this role presents us with additional challenges unique to mobile app development.

A. *Energy Availability Limitations*

If the objective of our work was to provide face recognition in a fixed location – say, one that we could bolt a desktop computer to – then energy availability becomes a non-issue: we could use as many different algorithms as we like and make them work as aggressively as possible in order to achieve the best possible result. But our objective is to perform face recognition on mobile handset devices, so therefore we are limited by the total amount of available energy being provided by the device's batteries or other energy generating devices. Therefore, the algorithms that we choose to use must find a balance between exhaustive statistical operations and power efficiency in order to be considered viable. We achieved this objective through our experimentation that aimed to discover the minimum effective training set size that is needed.

B. *Hardware Capability Limitations*

We must also take into consideration the hardware specifications of the device: specifically, what kind of imaging equipment is broadly available to us. Using a combination of very high resolution visible spectrum and infrared cameras would allow us to do sophisticated iris recognition and matching in order to perform identity verification, however the most readily available off-the-shelf handset device hardware of the present day can realistically only promise us a medium resolution camera, and maybe perhaps a simple rangefinder on newer devices, but not much else. Our method for performing face recognition must take into consideration that we are targeting off-the-shelf hardware that is broadly available, so therefore we have only so sophisticated of imaging devices available: they are not the cutting edge or the highest available resolution and therefore our method must be able to function accurately using very general imaging devices; working with an available image resolution not much higher than a common laptop webcam.

C. *Network Availability Limitations*

We also need to consider network availability as a capability limitation when performing face recognition on a mobile device. As mentioned in our considerations regarding energy availability, the sophistication of the system that we choose to use is limited by the computing and energy resources that are available on the device. Naturally, one may raise the suggestion that we could just perform data collection on the mobile device, then ship that data out over the network to be processed on a server somewhere, where power and resource utilization cease to be a concern. However, mobile data service availability is not guaranteed, and the central back-end server process that would perform the recognition could suffer a failure that could potentially affect all users of the system. Both of these failure conditions are completely unacceptable in the context of authentication. If the goal of this research is authentication and identity checking, then we cannot lock mobile users out of their devices because the network is not available or the back-end compute application is broken or offline. Authentication applications must be standalone and always available as long as the mobile device itself is working.

D. *Usability Considerations*

The notion that an authentication service must not be hampered by network unavailability or back-end service unavailability evolves into the fourth design challenge of our research: usability. The final product developed must be usable and easy to understand by the general population. When enrolling a user, we theoretically could capture thousands of images of their face from all different angles and build up a very accurate and comprehensive map of that user's statistically unique features. However, no user will want to sit for so long and have so many images taken of their face. Conversely, when authenticating we may run many different algorithms and try many different methods for determining identity, but if our face recognition software is built into an application that controls access to the mobile device, then users will not tolerate any unreasonable delay that prevents them from being able to authenticate successfully and then begin using the device.

V. Developing a Mobile Face Recognition System: Construction, Testing, and Evaluation

Using what we have learned so far about face recognition and the behavior of LBP with SVM systems, we developed a mobile application prototype designed to run on an "*off the shelf*" Android smartphone: the device contained no special imaging hardware other than the stock visible spectrum cameras, and a stock kernel that had not been modified in any appreciable way. We designed our application following the usability considerations we had ideintified: we aimed to build a screen locking application that is both convenient enough that it would be palpable by the average consumer, while remaining accurate enough to be considered relatively secure.

Our experimentation up to this point has determined that we can create an effective asymmetric SVM training set with as little as four positive samples of a subject to be enrolled and ten negative samples to create a complete training set. However, the next question we faced was in regards to the training protocol: we need to train the user on our system for this system to recognize them, by virtue of the needs of the machine learning software that we have elected to use. What are the best series of training scenarios that we could put the user in when we capture our positive training examples? LBP with SVM is capable – theoretically – of compensating for variable environment conditions, but we postulate that the ability of this system to tolerate for these variances is affected by the light conditions of the scenarios that we place the user in during the training phase. Therefore, we hypothesize that training protocol design is an important consideration in optimizing system accuracy.

Now, when we refer to "*training scenarios,*" we are referring to the light conditions that are present when each of the positive images for enrollment are captured. By the nature of how the SVM algorithm operates, it would make sense that we would want to present the model during training with as many extremes of light conditions as we possibly can. But how much of a difference does it really make? It creates a connundrum for us: we want to build an application that is usable, and making a user do something other than stand in one spot during enrollment is conunterintuitive to that goal. However, if it is justified that user identification becomes more accurate, then perhaps it might be worthwhile to utilize a more complex training protocol.

We therefore devised two different training protocols in order to test our hypothesis. Our fist protocol, the control protocol, used only one training scenario for all four training images. The scenario was inside of a windowless office: a plain white background with white flourescent light positioned directly overhead (Figure 8). The second training protocol aimed to evaluate whether the inclusion of variable light conditions would lead to an increase of system accuracy. We created four different training secnarios:

Figures 8, 9, 10, & 11: (Clockwise from top left) Training scenario one: an office with overhead lights and no windows. Two: a hallway with offset light and no windows. Three: a hallway with both artificial and natural light. And four: a mezzanine with a large curtain window.

1. The first scenario was the control scenario (Figure 8).
2. A scenario was created that used a dark backdrop with artificial light positioned at an offsett (Figure 9).
3. A scenario was created that used artificial light positioned at an offset combined with a backdrop of natural light channeled through a hallway (Figure 10).
4. A scenario was created that used natural light projecting through a large glass window wall that was located behind the camera (Figure 11).

Since our experiments were performed in the middle of winter in the Northern hemisphere, we noticed that the properties of the available natural light in the fourth scenario changed noticeably throughout our testing over the course of the afternoon, as the sun rapidly traversed across the sky.

A. Evaluation of Training Protocols

Our testing was conducted as such: for each test subject that we evaluated, the test subject would either be trained using the control protocol, which trained the user in the first scenario only, or they would be trained using the experimental protocol, which would train the user in all four of the scenarios that we had set up in and around our laboratory. The user would then be tested by evaluating how well the system performed in each of these four scenarios: users would be taken to each of these four locations and asked to use the device in our software's "authenticate" mode (Figure 12). In addition to performing "positive" tests, where we would expect the system to correctly identify the subject as being the person who enrolled, some participants were also asked to participate in "negative" tests, where we would train the system with one user and authenticate with another user, expecting that the system would correctly identify the subject as not being the enrolled person.

Of course, the biggest problem that we faced in this sort of realistic field testing was finding willing participants who wanted to be a part of the study. In total, we were able to coax and encourage nine participants to be a part of this preliminary experiment. We ran four positive control tests, two negative control tests, four positive experimental tests, and two negative experimental tests. Altogehter, 48 datapoints were collected, however only 47 datapoints were viable for use due to a software runtime error that occurred in one of the evaluations that resulted in the test being unable to continue.

What we found was such: 21 of the 23 control datapoints that were valid returned a "correct" result: the user was positively identified during positive tests or correctly denied during negative tests. This equates to a control accuracy of 91.3%, in line with what we expected to see, given the results of our earlier Caltech database experimentation. In multi-scenario training, 24 of the 24 valid datapoints that were collected returned correct, yielding an accuracy of 100%. Now naturally, we do not really consider this to be a conclusive result; we only had nine participants in total and the test was rather primitive in its construction. However, what we do believe is that this result implies that an LBP with SVM face recognition system can be made more accurate if a training protocl is utilized that exposes users to a variety of different light conditions both natural and artificial in their source, allowing this face recognition system to achieve what appears to be reasonable environment tolerance.

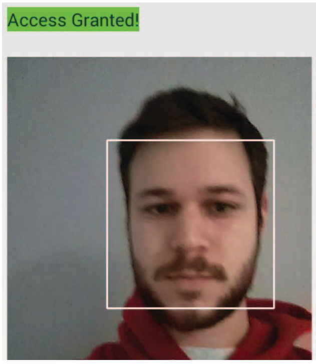


Figure 12: Sample output from our field evaluation software.

VI. Discussion and Conclusion

It is very interesting work for us to be studying face recognition. It is a technology whose capabilities are often overstated and its operating principles generally misunderstood by the public at large. Many assume that face recognition is a single, homogenous technology that has over the years become so accurate that it risks causing severe social problems for society as a whole. But this caricature could not be further from the truth.

The reaility is that face recognition is still primitive, desipite the impressive pedigree of historical research efforts that have been published. It is a very wide range of very different technologies that all take their own philosophical approaches to computerizing a mysterious neurological process.

In one of the strange and weird connundrums of face recognition, we need to recognize that the problem of environrment tolerance is not even really what it seems. When we as humans look at someone's face, we aren't really looking at their face, as counterintuitive as that sounds. What we are looking at are subatomic particles of light that have been reflected off of someone's face and into our eyes, where it gets converted into an electrical signal and then gets processed through highly evolved neurological circuitry that is nowhere near entirely understood by science. These photons are not all the same: they have varying wavelengths, varying intensities, and their origins come from all different directions, depending on what kinds of light sources are nearby. And we as humans can mentally compensate for that. But we do not really understand at a scientific level how exactly our brains accomplish this, so emulating this mechanism with computer software becomes an intensely difficult challenge.

We have found that feature analysis systems like LBP are much more tolerant to changes in light conditions because their fiduciary data are only influenced by local points immediately around them and not by the entire face area as a whole, minimizing corruption by environment variability. Our prototype software is developed around this theory, and our preliminary experimentation seems to positively indicate that we are able to achieve acceptable accuracy with real users that are exposed to real multi-environment conditions.

But our work is not finished yet. We have only marginally proved that LBP is capable of providing total environment tolerance under all possible conditions. Our prototype software was primitive and our sample user base was small, but the results we obtained were very encouraging. We believe that LBP with SVM is an ideal solution for creating a lightweight face recognition system that is tolerant enough to environment changes such that it may become viable for use in mobile authentication applications. We intend to continue to develop this technology and expose it to harsher and more random light conditions in order to find where the limits of this system lie. We hope to give face recognition technology its *flying car* moment; to bring us closer to general-case face recogition becoming a powerful and flexible tool in the biometric authentication toolbox.

Acknowledgment

This work was sponsored by the Air Force Research Laboratory (AFRL). Distribution A. Approved for public release; distribution unlimited (Case No. 88ABW-2015-2589). The authors wish to thank AFRL in conjunction with the Griffis Institute of New York for providing us with laboratory facilities during a portion of this research.

The authors also wish to thank all of the volunteers at Temple University who agreed to participate in our field trials.

References

[1] M. Nixon, "Eye Spacing Measurement for Facial Recognition," SPIE vol. 575: Applications of Digital Image Processing VIII, pp. 279-285, 1985.

[2] I. Sobel, G. Feldman, "A 3x3 Isotropic Gradient Operator for Image Processing," presented at the Stanford Artificial Intelligence Project, 1968.

[3] P. V. C. Hough, "Method and Means for Recognizing Complex Patterns," U. S. Patent No. 3,069,654.

[4] J. Canny, "A Computational Approach to Edge Detection," IEEE Trans. on Pattern Analysis and Machine Intelligence, 8(6), pp. 679-698, 1986.

[5] F. Timm, E. Barth, "Accurate Eye Centre Localisation by means of Gradients," Proc. of the International Conference on Computer Theory and Applications (VISAPP), Vol. 1, pp. 125-130, 2011.

[6] T. Ojala, M. Pietikaeinen, D. Harwood, "A Comparative Study of Texture Measures with Classification based on Feature Distributions," Pattern Recognition 29, pp. 51-59, 1996.

[7] L. Wolf, T. Hassner, Y. Taigman, "Descriptor Based Methods in the Wild," Real-Life Images Workshop at ECCV, 2008.

[8] T. Ojala, M. Pietikaeinen, T. Maenpaea, "Multiresolution Gray-scale and Rotation Invariant Texture Classification with Local Binary Patterns," IEEE Trans. on Pattern Analysis and Machine Intelligence 24, pp. 971-987, 2002.

[9] T. Ahonen, A. Hadid, M. Pietikaeinen, "Face Recognition with Local Binary Patterns," IEEE Trans. on Pattern Analysis and Machine Intelligence 28, pp. 2037-2041, 2006.

[10] University of Massachusetts, "Labeled Faces in the Wild Database," `http://vis-www.cs.umass.edu/lfw/`

[11] Itseez, "OpenCV Open Source Computer Vision Library," `http://opencv.org`

[12] U. S. National Institute of Standards and Technology, "The Facial Recognition Technology (FERET) Database," `http://www.itl.nist.gov/iad/humanid/feret/feret_master.html`

[13] California Institute of Technology, "Computational Vision Frontal Face Dataset," `http://www.vision.caltech.edu/html-files/archive.html`