Facial Recognition

**Project Members:** *Haider Syed & Travis Peters*

**Project Advisor:** *Lorenzo Torresani*

Dartmouth College Computer Science – Computer Vision (CS 183)

October 29th, 2013

# Introduction

## 1.1 Background

This project was inspired by work being done in the mHealth Security & Privacy lab to address authentication with a mobile medical device. Some of the members of the lab were working with students from another school to construct a low-cost and mobile device known as a Spirometer with a camera installed on it.

This background information helps to provide insight into the motivation for the problem but it should be clear that our project aims to address *only* the problem of recognition from an image.

The aforementioned device has yet to determine the exact composition of the mobile device/camera position, therefore our project doesn’t have a representative dataset to validate our ability to perform accurate recognition. In light of this, we will not consider hypothetical scenarios from a potential dataset and we will focus on addressing four main problems in the realm of facial recognition:

1. Complex backgrounds
2. Varying subject pose
3. Varying subject scale
4. Varying illumination conditions

In this paper we present the current state our implementation and demonstrate how, at this point, we have begun to address the issues mentioned above. We also propose a vision of the modifications we will be making to our implementation in order to obtain better recognition results, and a justification for why we believe these modifications will improve the accuracy of our facial recognition system.

## 1.2 Assumptions

The following assumptions were made in order to control the scope of our project and keep our efforts as focused on the problem of recognition in the context of a system that would be used by people using the mobile medical device indicated above:

* The input image represents someone who is in the database that we intend to search.
* We only account for pose variations up to 60 degrees (this is a limitation of the SIFT algorithm which our recognition system utilizes).

# Methodology

We approach the problem of face identification for an image by breaking it down into three sub-problems:

(1) Segmenting the face from the background, (2) extracting local feature descriptors, and, (3) using a nearest-neighbor clustering to identify a particular image/set of images that best matches the input image.

**Note:** *the images that the input image is compared against have already been segmented from their respective backgrounds and have had local feature descriptors extracted – in the case that multiple images exist for any one person, a concatenation of their feature descriptors compose a set which is used to represent that person.*

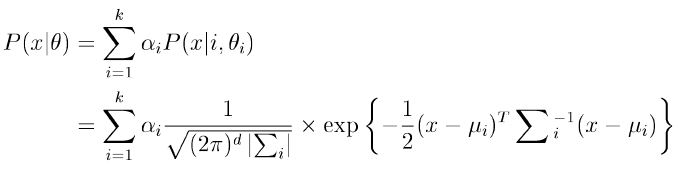
In this section, we (1) propose an implementation to tackle the aforementioned problems, and (2) present our current implementation.

## Proposed Approach

We have been continuously researching methods that lend themselves well to solving the sub-problems that we have previously identified. Below, we briefly introduce these methods and comment on their relevance to this project.

### *Gaussian Mixed Model (GMM)*

To be able to account for complex backgrounds in the images, we try to separate the face pixels from the background. It has been shown that skin pixels cluster in a part of the color space even for people of different races. Using this information, it is possible to compute a probability that a pixel in a given image is a skin or non-skin pixel. A Gaussian distribution can be used to model the probability that a pixel is skin and non-skin, respectively. However, using multiple Gaussians to model the probability distribution is able to better account for inter-class variance and for multiple sources of variance (such as illumination in the image) [7]. The probability distributions used in the Gaussian Mixed Model is shown below:



This model is the same as the *Single Gaussian Model* presented in section 2.2.1, so please see that section for specific details about how the model works. The difference between the models is that the GMM uses multiple Gaussians to model the color space while the Single Gaussian Model uses a single Gaussian distribution.

### *cSIFT*

SIFT is robust to translations and rotations but not illumination changes. An algorithm known as cSIFT [8] is a modification of the SIFT algorithm and is more robust to illumination changes. The algorithm uses a Fresnel reflectance model to compute a pseudo-color space from the RGB space. The resulting image has a single value for each pixel. These values are used to compute SIFT descriptors instead of grayscale values. As the values are established in a way that makes them invariant to changes in illumination the image, the resulting descriptors are supposedly invariant to the image. We are in the process of implementing this algorithm and will present it in detail for the final submission.

### *kNN*

Each person in our database has multiple images (for now we assume 3). For a single person, we compute the feature descriptors for each of the 3 images and concatenate it into a single matrix. This matrix now represents all the features identified in all 3 images for that person. We then compare each feature in the probe image to every feature in the feature set of every person in the database, by computing the Euclidean distance between the features. For each feature in the probe image, we find the best matching feature set in the database. We then identify the feature set in the database that has the most matches to features in the probe image – the identified set is the match our algorithm returns. We call this algorithm *consolidated kNN (ckNN)*.

## Current Implementation

While we have previously identified our ideal solution, some of the methods discussed thus far are fairly complex and we were unable to implement them in their entirety before the milestone. Below, we present the current state of our implementation and any relevant details.

### *Single Gaussian Model*

The first problem we aim to solve is the issue of having the face amongst a complex background. If we simply try to extract feature descriptors from an image with a complex background, there is a high likelihood of picking up features in the background, which are obviously not representative of the person we are trying to identify. Thus, we have implemented a Single Gaussian Skin Model to allow us to train a model which can determine if a given pixel in an input image is more likely to be a skin pixel or a non-skin pixel.

Our implementation currently breaks the training into two parts: part one allows a user to open an image and interactively define contour regions that consist of skin; part two opens the same image but allows the user to now define non-skin regions. This is all the information needed to construct our Gaussian Probability Model that, given a pixel from an input image, returns the probability that a pixel with this particular value is a skin/non-skin pixel – thresholding on all of the pixels in an input image results in a raw image in which skin regions have been identified. The training process must only be computed once as the values should hold even for other images. We have set it up so that the training can be done using multiple images in order to better describe the skin and non-skin color spaces, respectively.

For a probe image, the model is applied to each pixel to determine whether or not the pixel is a skin pixel. The image outputted from this step may miss some parts of the face and, furthermore, is not supposed to pick up the mouth and eyes. Therefore, the identified face region will have holes and may not be contiguous. We apply some post-processing steps to get a contiguous face region and to get rid of false matches in the background. We start by eroding the image with a small circular kernel to get rid of small regions of false matches in the image. We then fill holes in the image using morphological reconstruction. We then isolate the largest contiguous segmented region by area, and define it as the face of the person we want to recognize.

Preliminary results from our implementation of a Single Gaussian Model are displayed/described in Section 3.

### *SIFT (feature descriptors)*

Once we have an image that is segmented from the background, we obviously need a way of representing the person that we are trying to recognize. For this, we use the SIFT feature descriptors which allows us to extract scale invariant features from an image.

We attempted to implement our own version of SIFT based on the description of the algorithm found in David Lowe’s paper [3] and Andrea Vedaldi’s notes [4], but have had problems with the correctness of computing the scale spaces. Therefore, for our milestone results we have utilized Vedaldi’s implementation of SIFT in MATLAB [4].

### *kNN*

Feature classification, though not really a Computer Vision problem, is critical for our overall facial recognition system. Upon recommendation from Lorenzo Torresani, we’ve decided to use the K-Nearest-Neighbor classifier as it yields good results so long as there are meaningful feature descriptors. [2]

The implementation of the kNN component was straightforward given our needs: since any given person in the database has multiple images, we first grouped the feature descriptors for each image into a set by concatenating them into a single feature-matrix, whereby this feature matrix now represents a single person. We then proceed by computing *L2* norms between each feature descriptor for the input image and each feature descriptor for all the existing persons in the database. By looking at only the nearest-neighbor for each feature descriptor in the input image, we can find an identity which has the most commonality. When we locate the person having the most matching correspondences with the input image, we declare the person in the input image to have that identity.

Preliminary results from our implementation of the kNN classifier are displayed/described in Section 3.

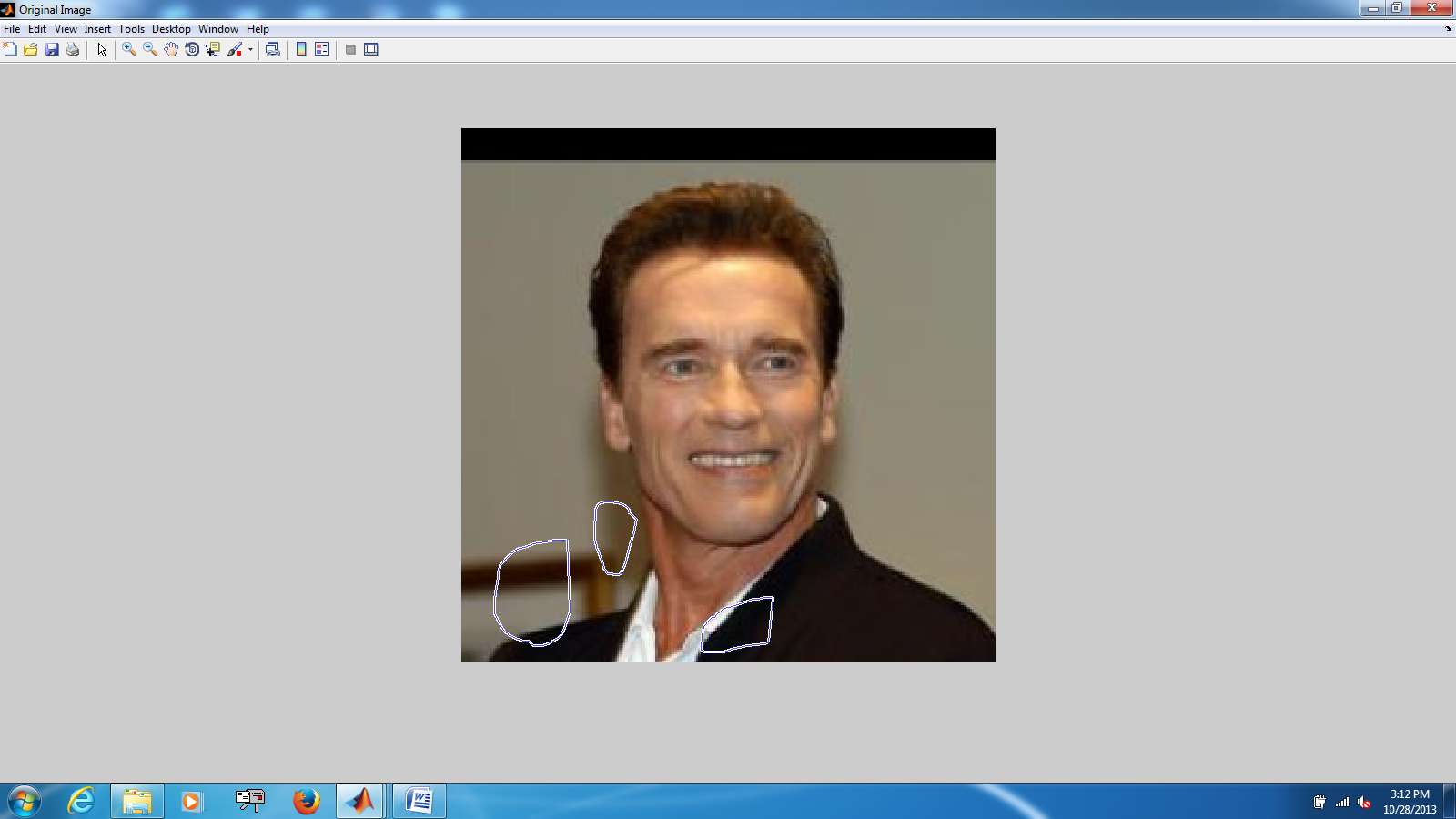
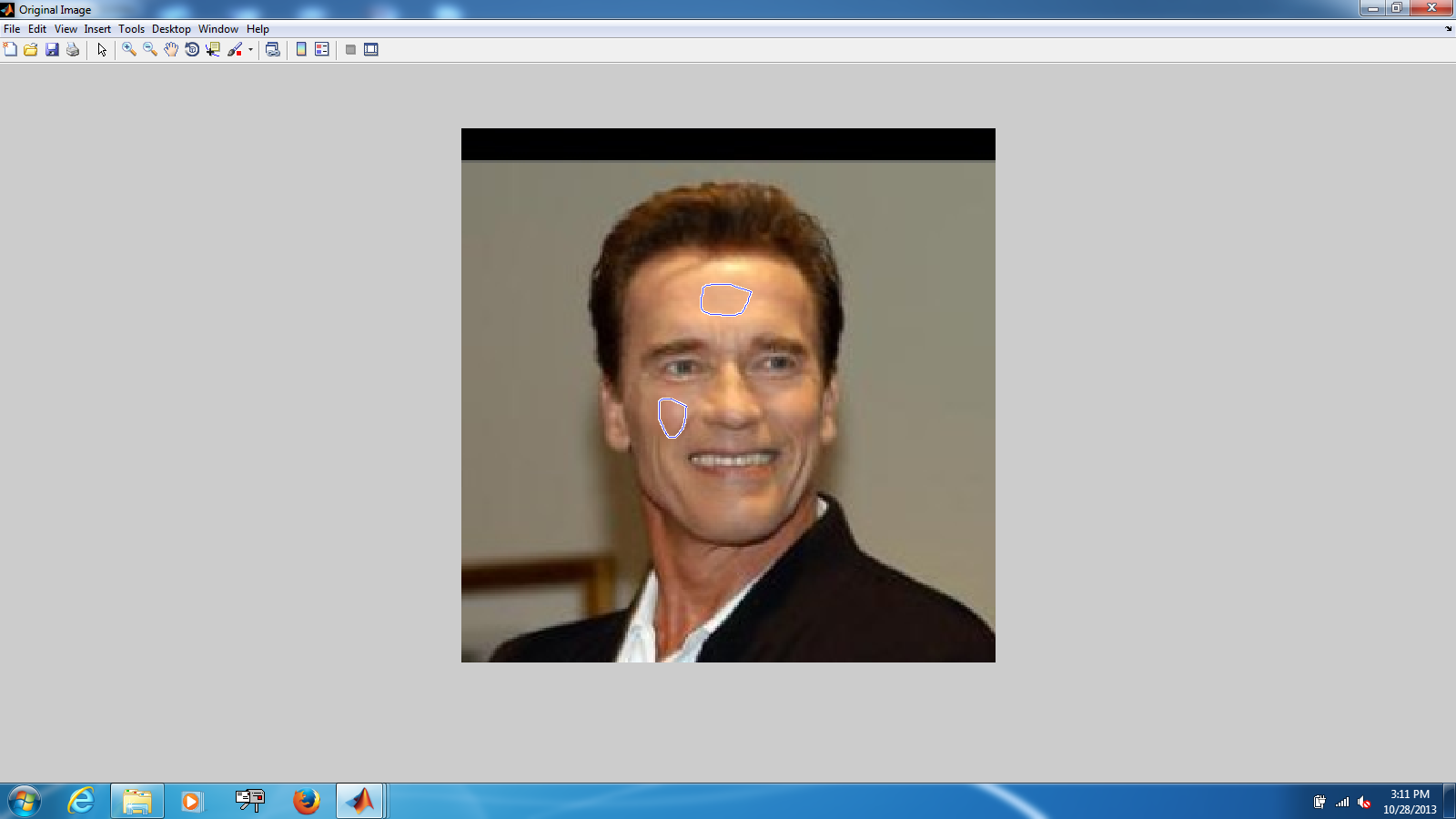
# Experimental Results

## Dataset

Currently we are using images from the *Faces in the Wild* database [5] and Aberdeen [6] for testing. We use images from *Faces in the Wild* in order to test the accuracy of our Single Gaussian Skin Model and its ability to detect faces amongst a complex background. We use the Aberdeen [6] dataset to test the implementation of SIFT that we are using (and the kNN classification stage), since SIFT is not invariant to illumination differences and is not accurate when there are variations in illumination conditions. We have tried using SIFT on a few images from *Faces in the Wild* and have not found it to work very well.

## Testing Single Gaussian Skin Model

Figure 1 shows our training stage for the Single Gaussian Skin Model that we have implemented. As mentioned in Section 2, the user can simply draw contours directly on an image and identify skin/non-skin regions. For all the results shown in this paper, the complete training set is shown in Figure 1. Although we could use further training, we found that using this small sample set gave us very good results. We were quite surprised that, given such a simple training phase, our model was able to correctly identify regions of skin surprisingly well!

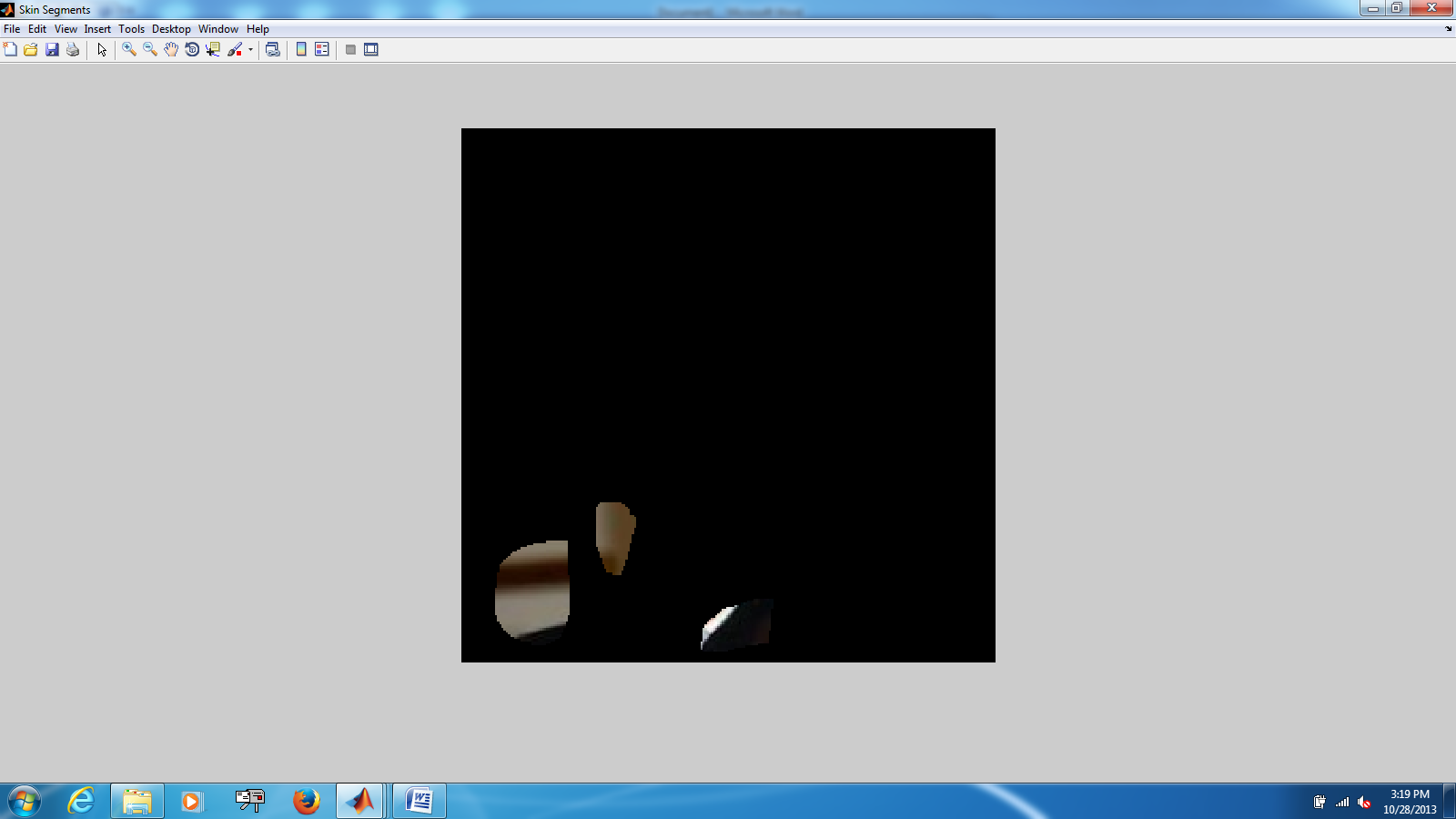
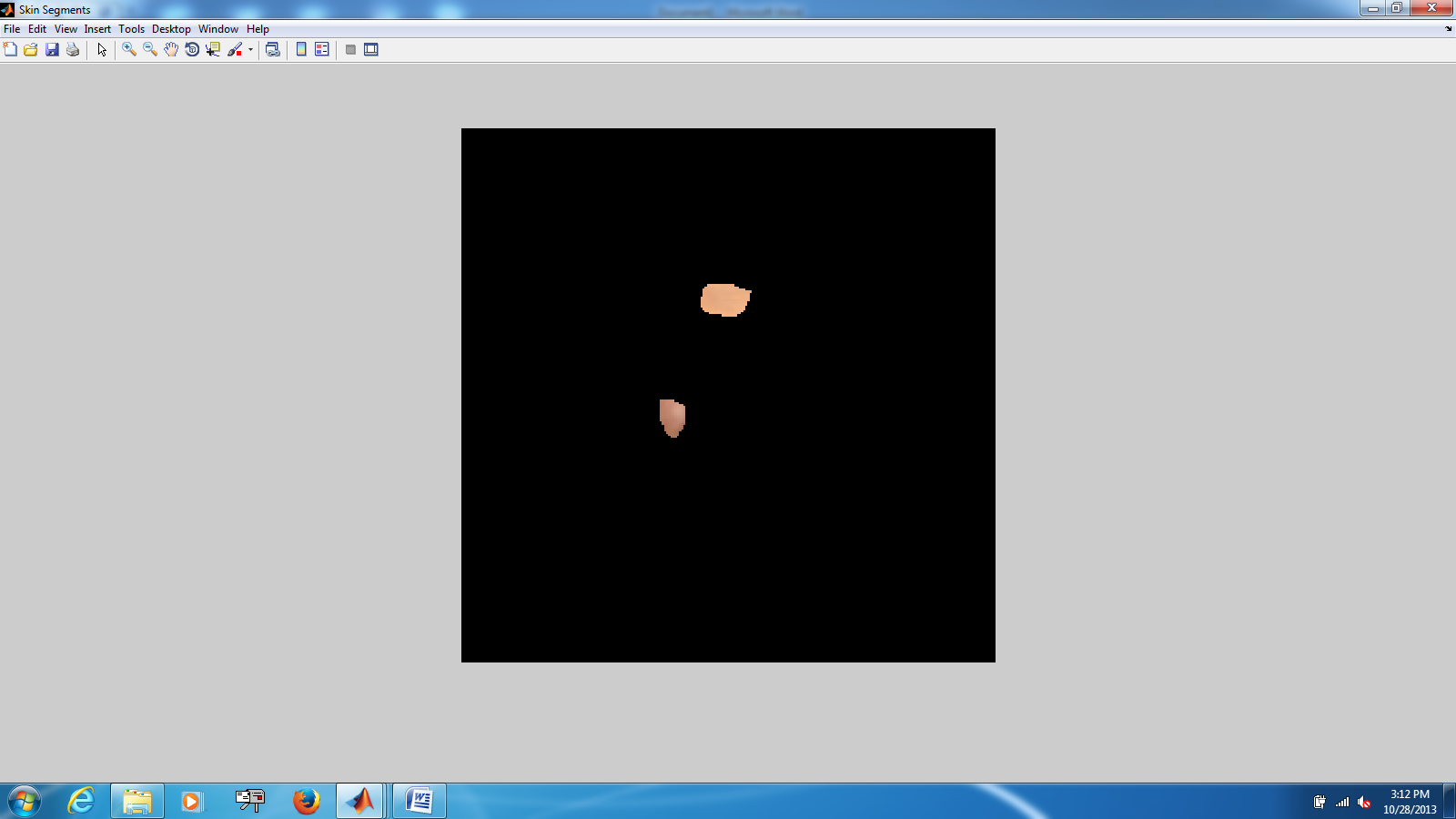
 

Figure 1: (Left) Non-skin training of Single Gaussian Model and (Right) training of skin model for Single Gaussian Model.

Figure 2 shows the intermediate steps of using the Gaussian Skin Model to detect skin regions on an input image (1st image). The second image shows the identified skin pixels directly using the Single Gaussian Model; as expected the eyes, mouth and nostrils are not picked up as skin. The third image shows the result of after our post-processing steps (erosion/filling of holes) in order to reconstruct contiguous skin regions. The last image in Figure 2 shows the bounding box that was created based on selecting the largest contiguous skin region in the image to be the face.

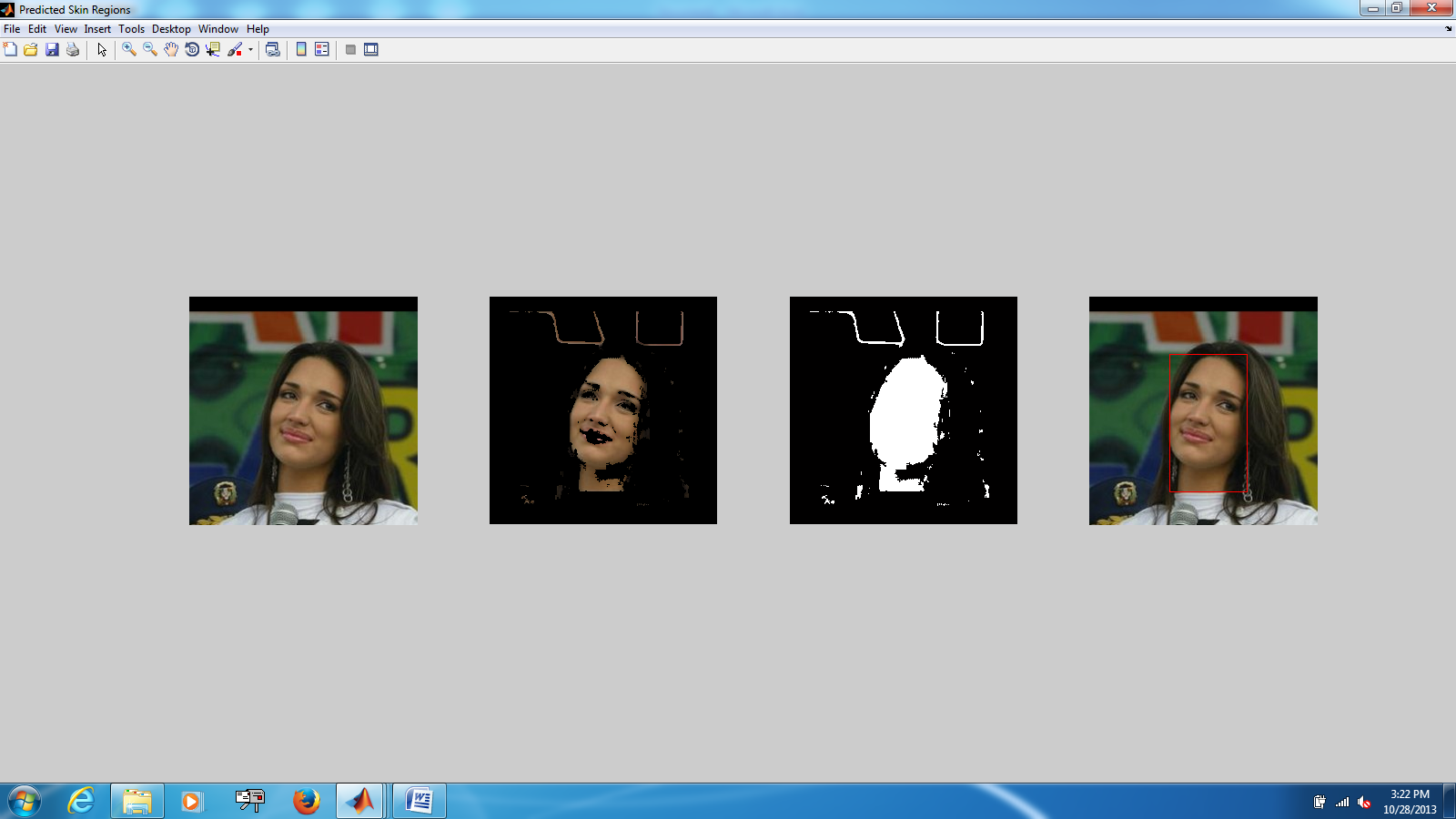


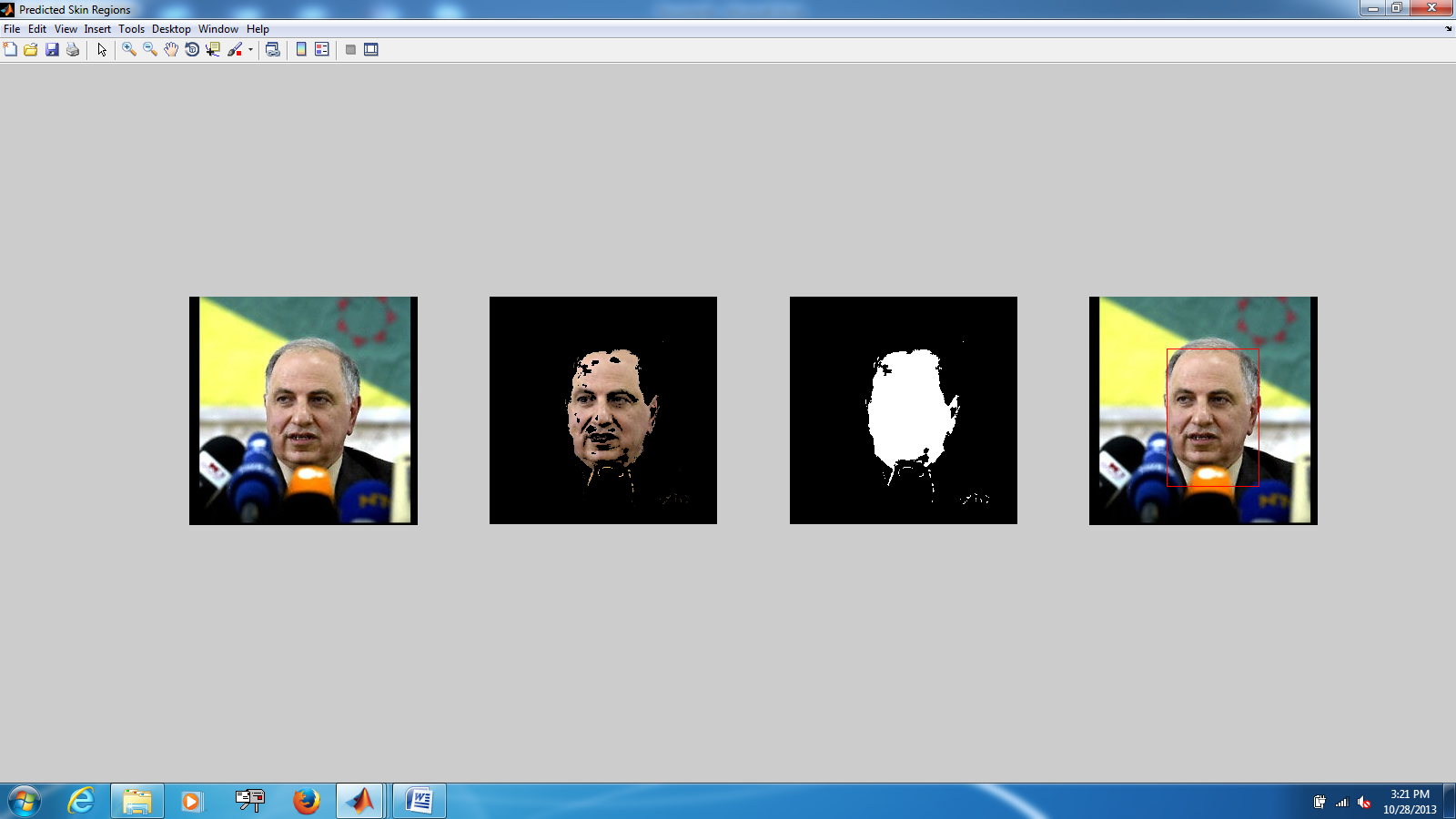
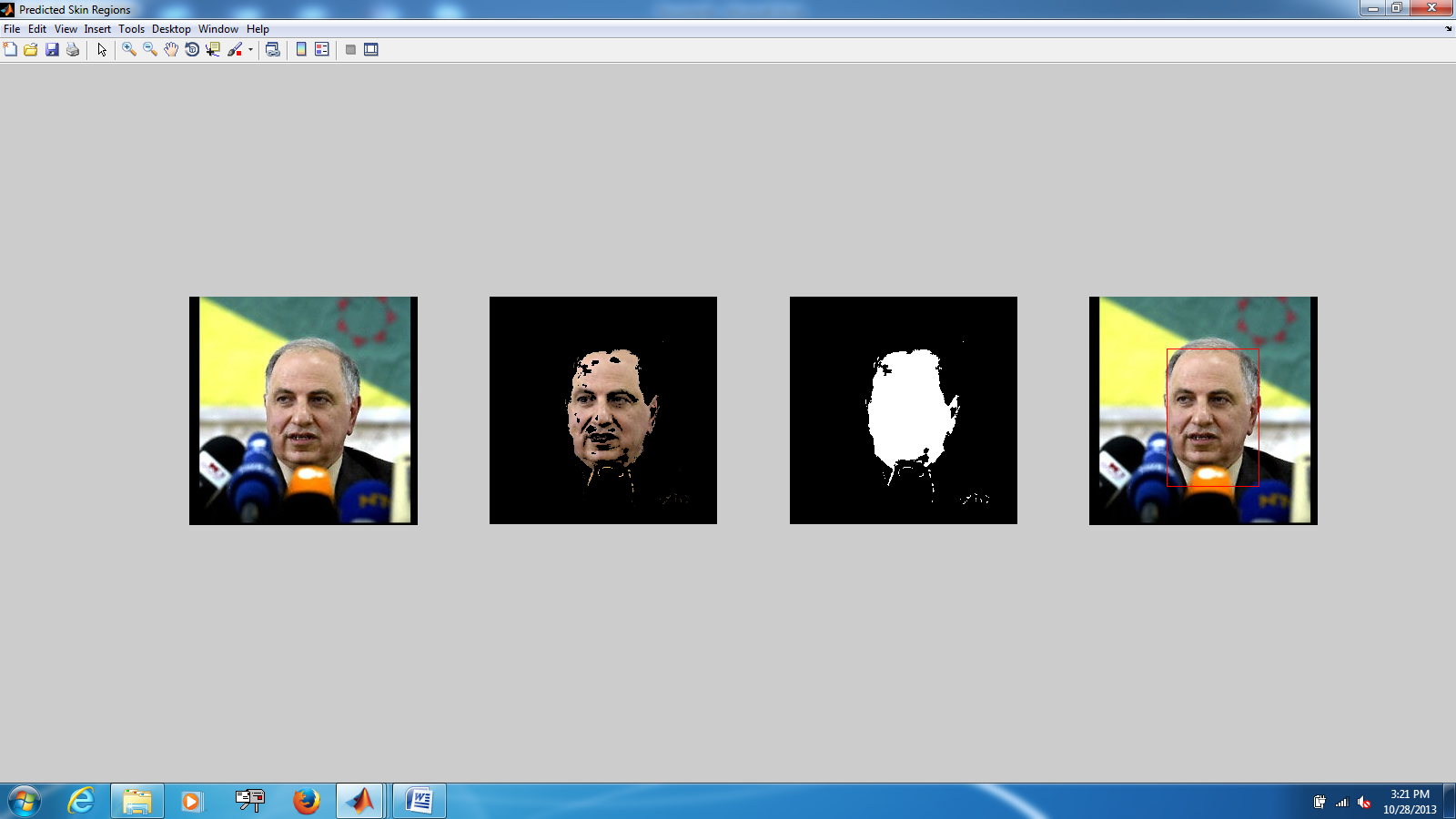
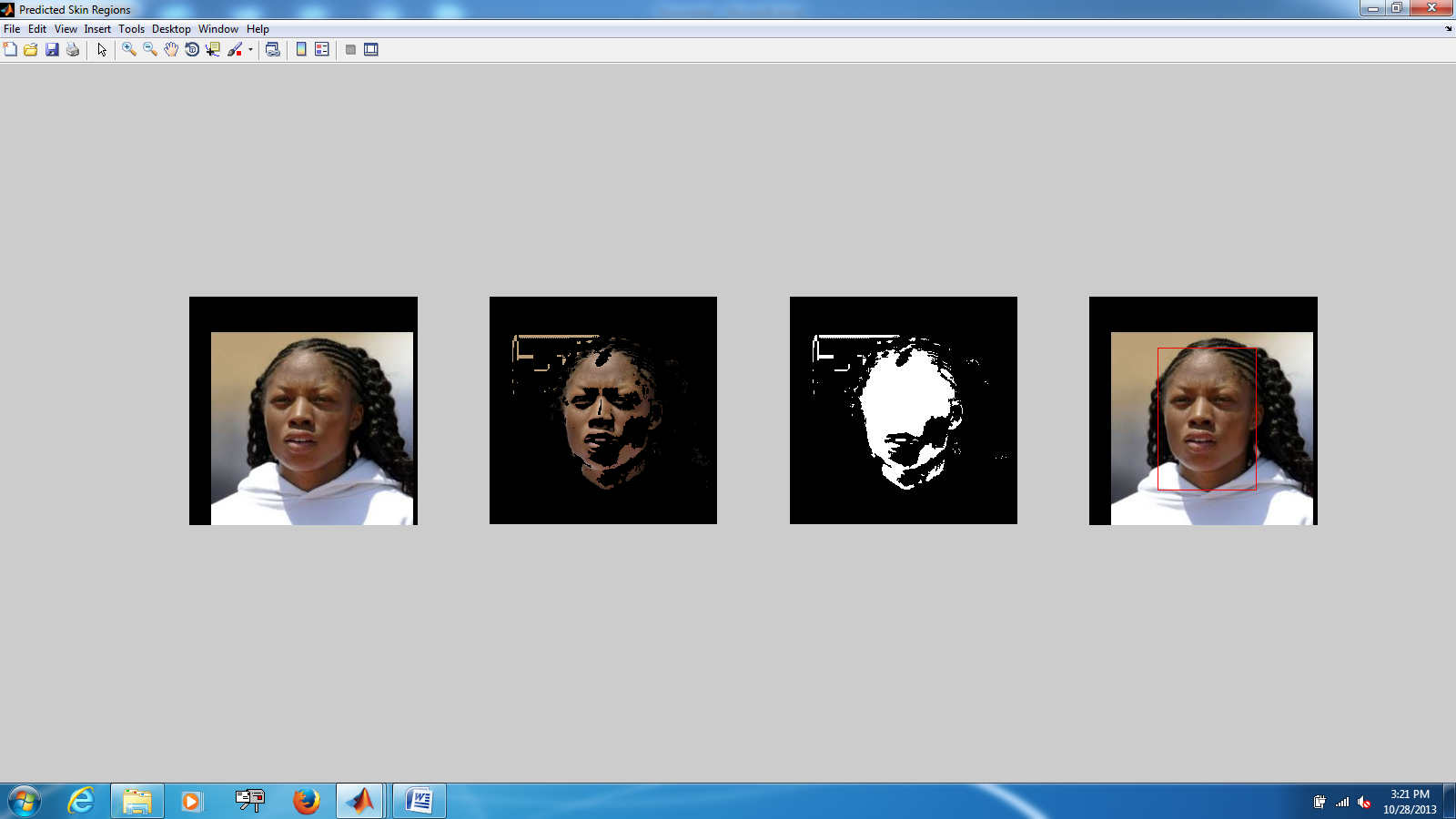
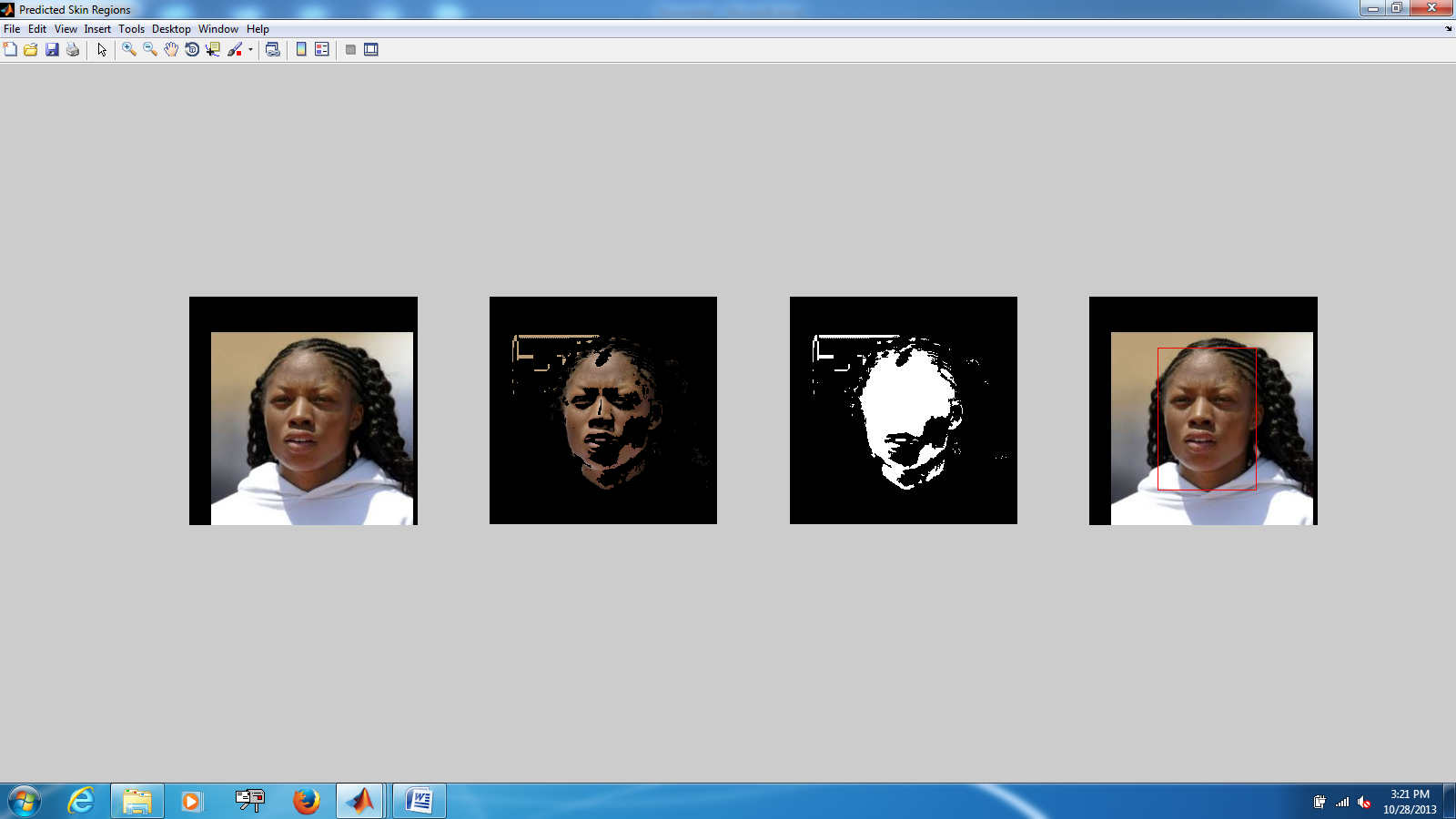
Figure 2: Intermediate steps described in Single Gaussian Model skin segmentation.

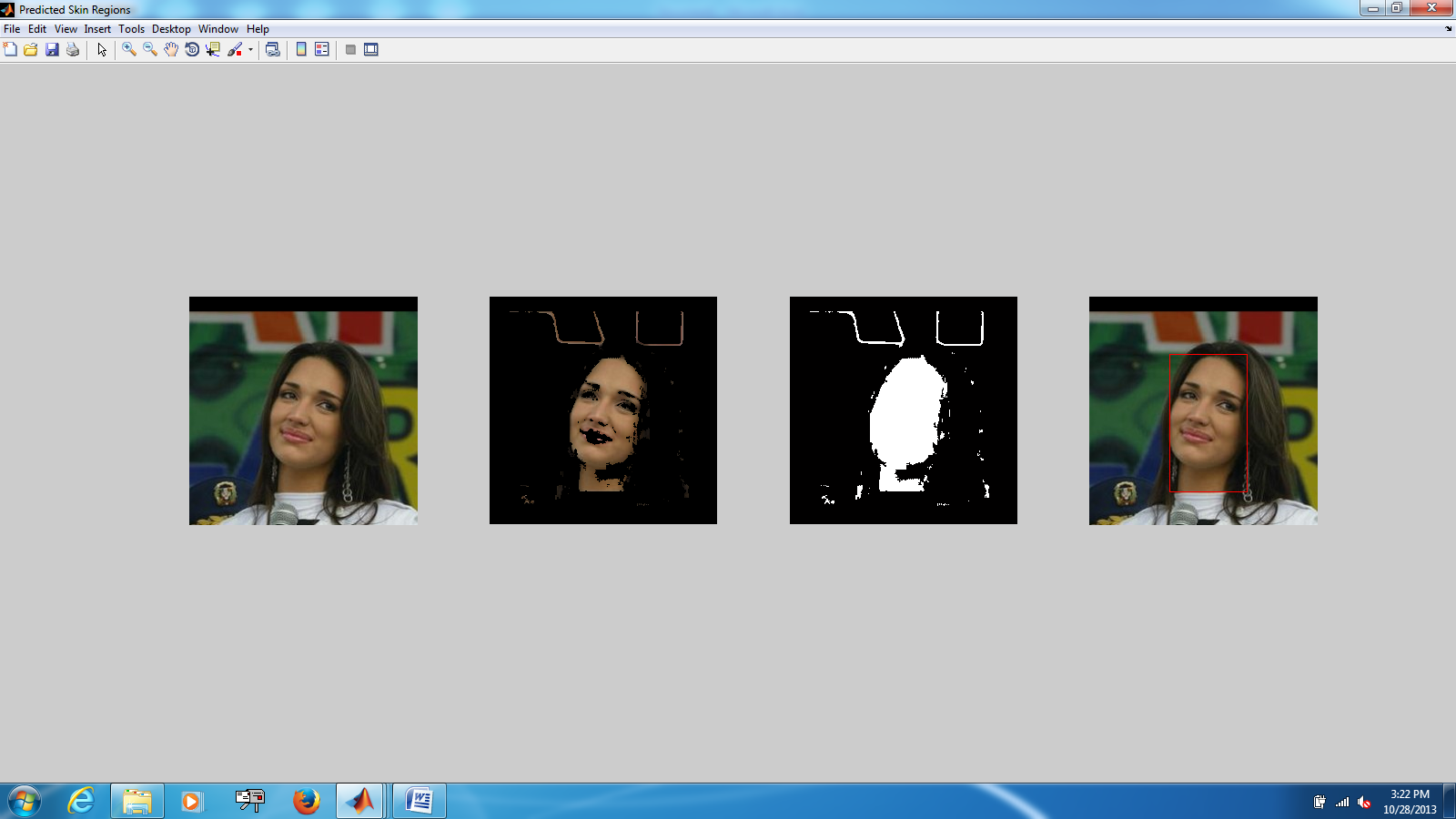
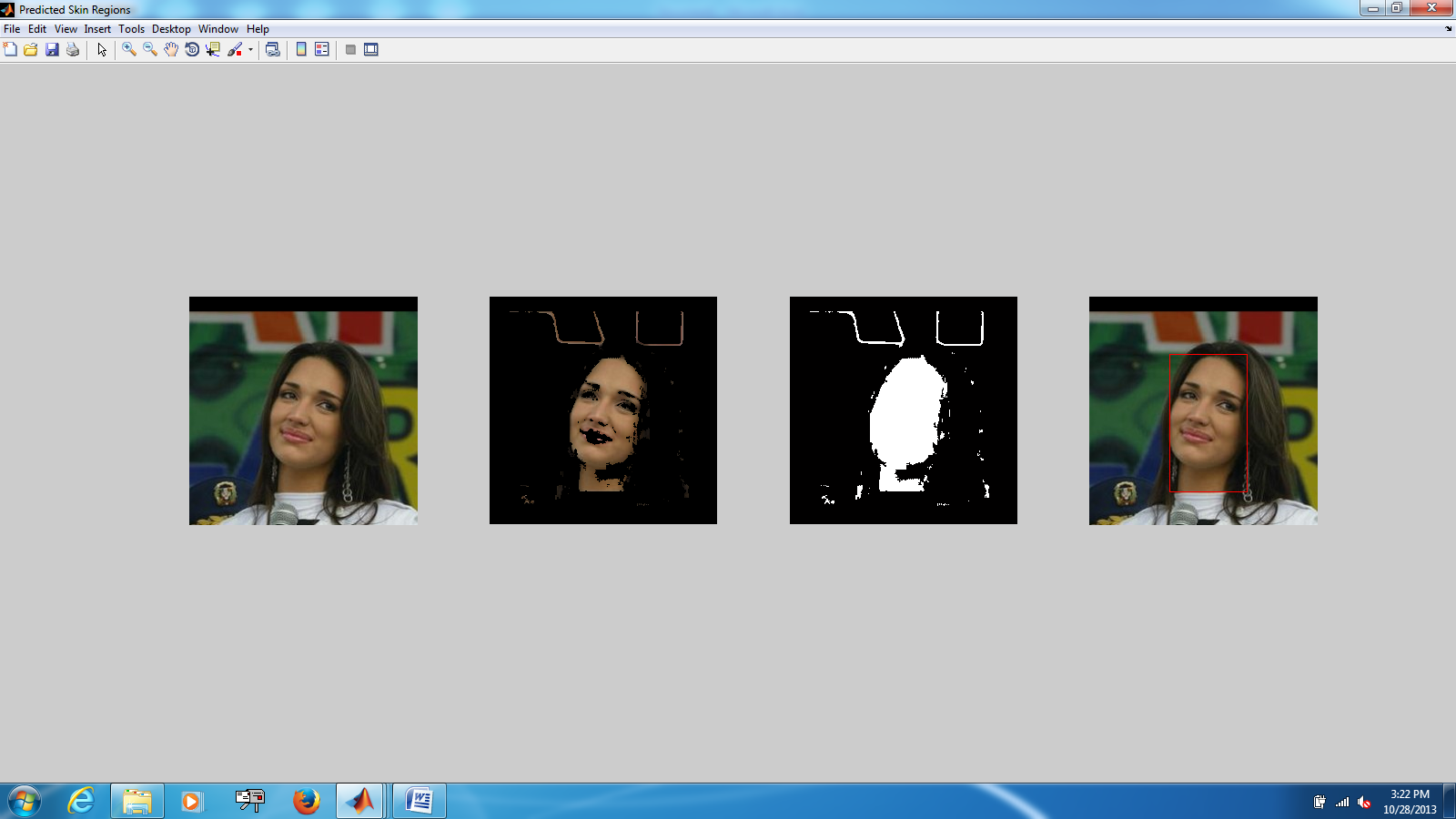
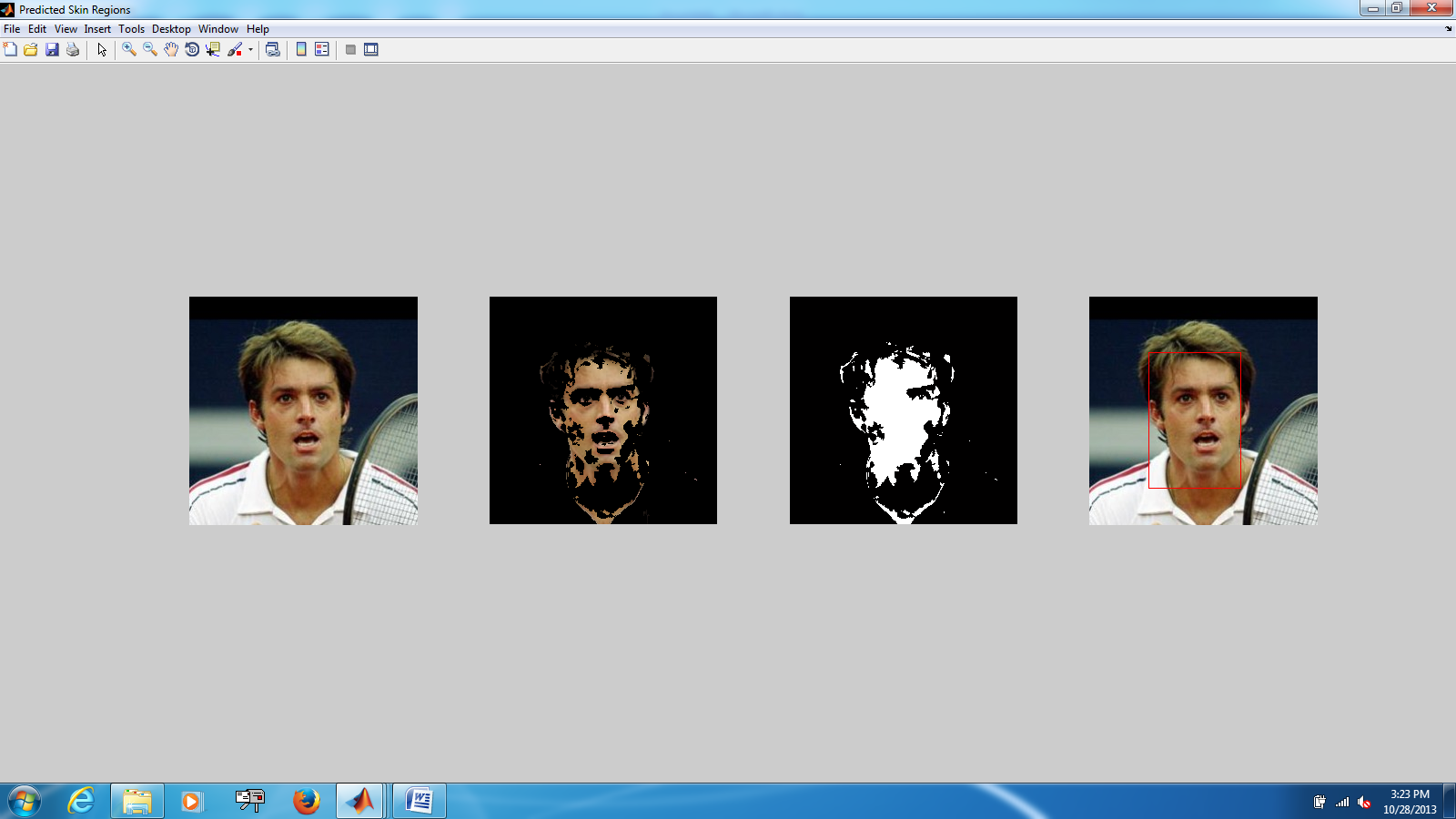
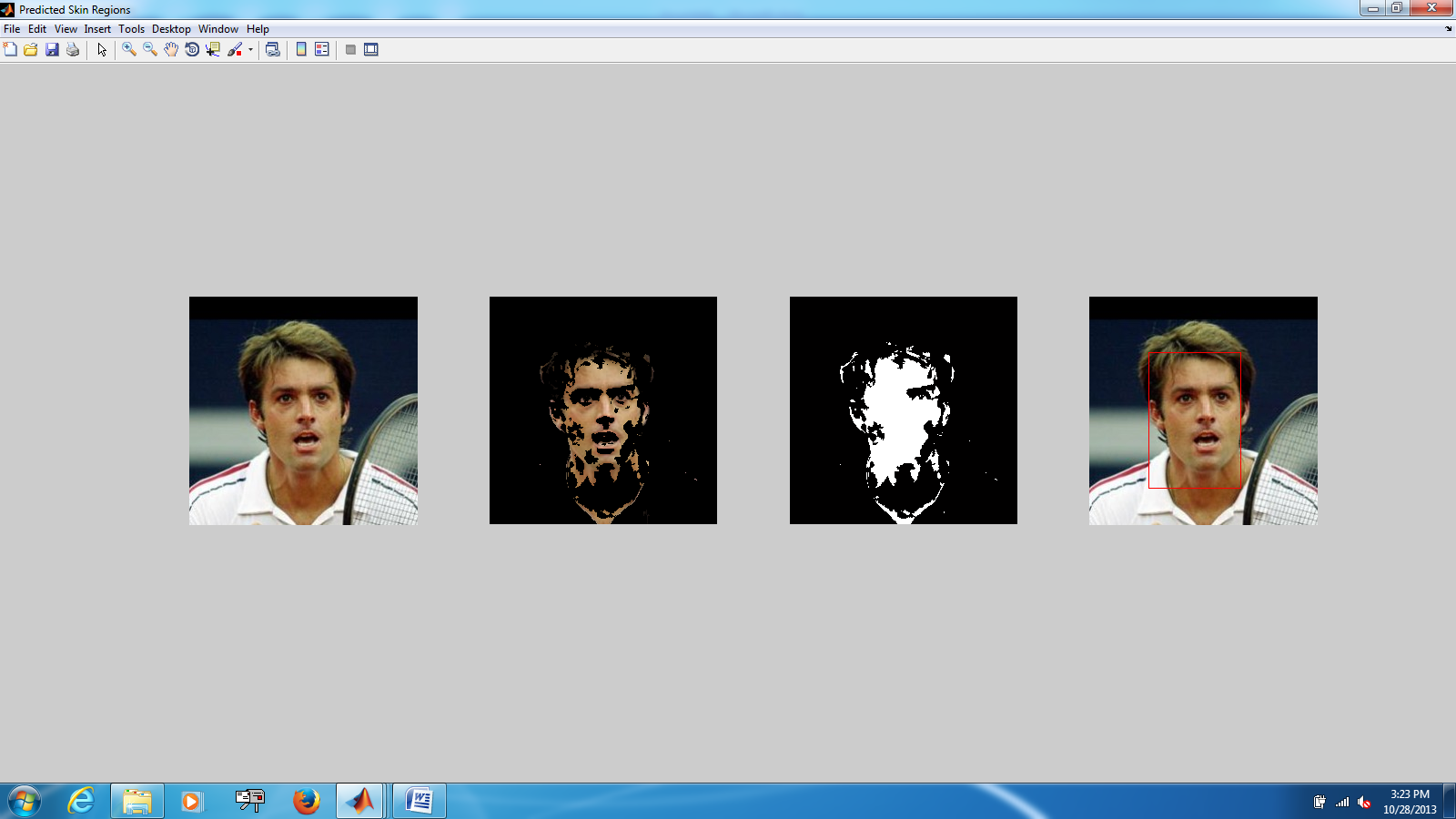
**Testing on the Aberdeen Database**

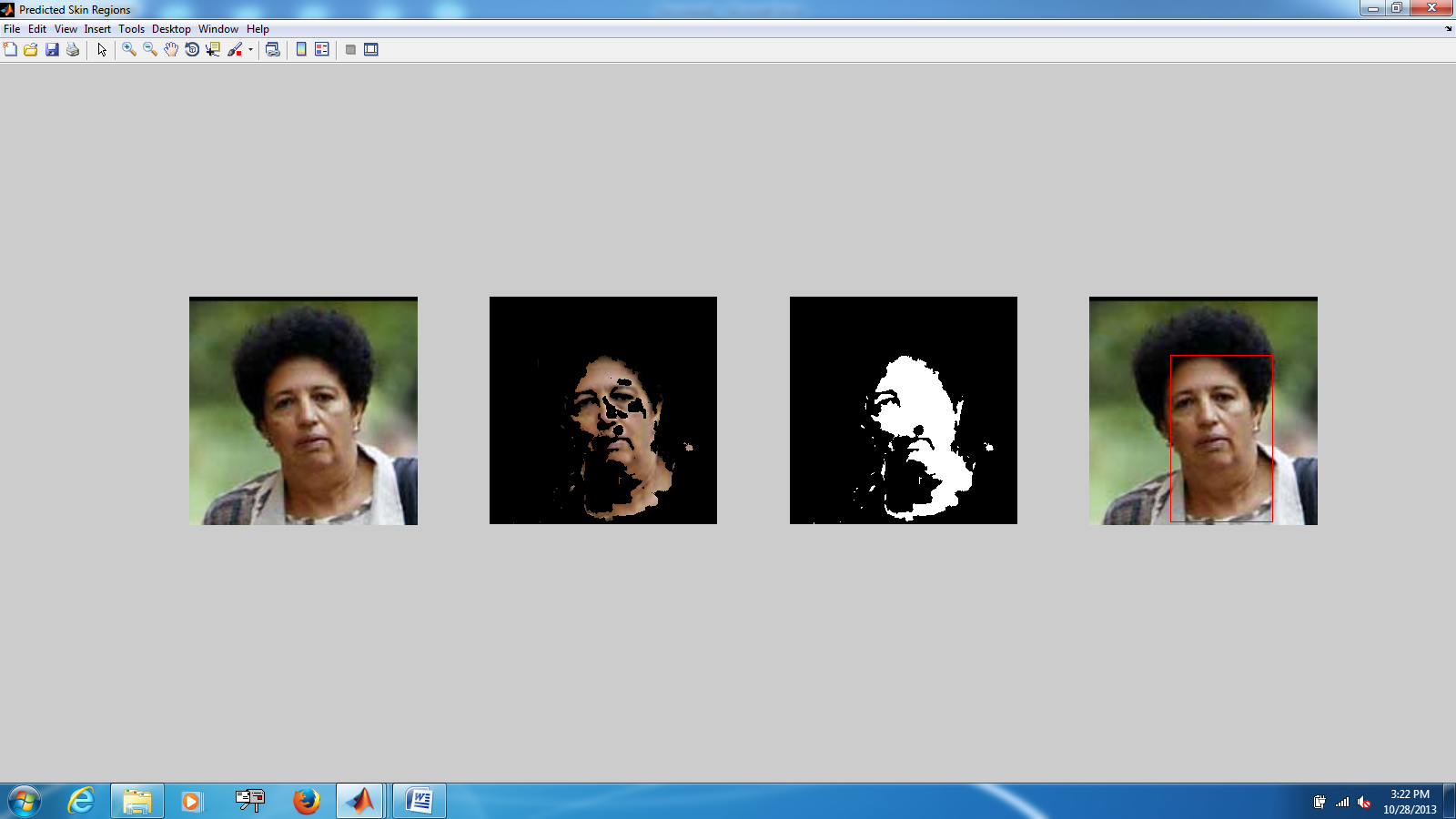
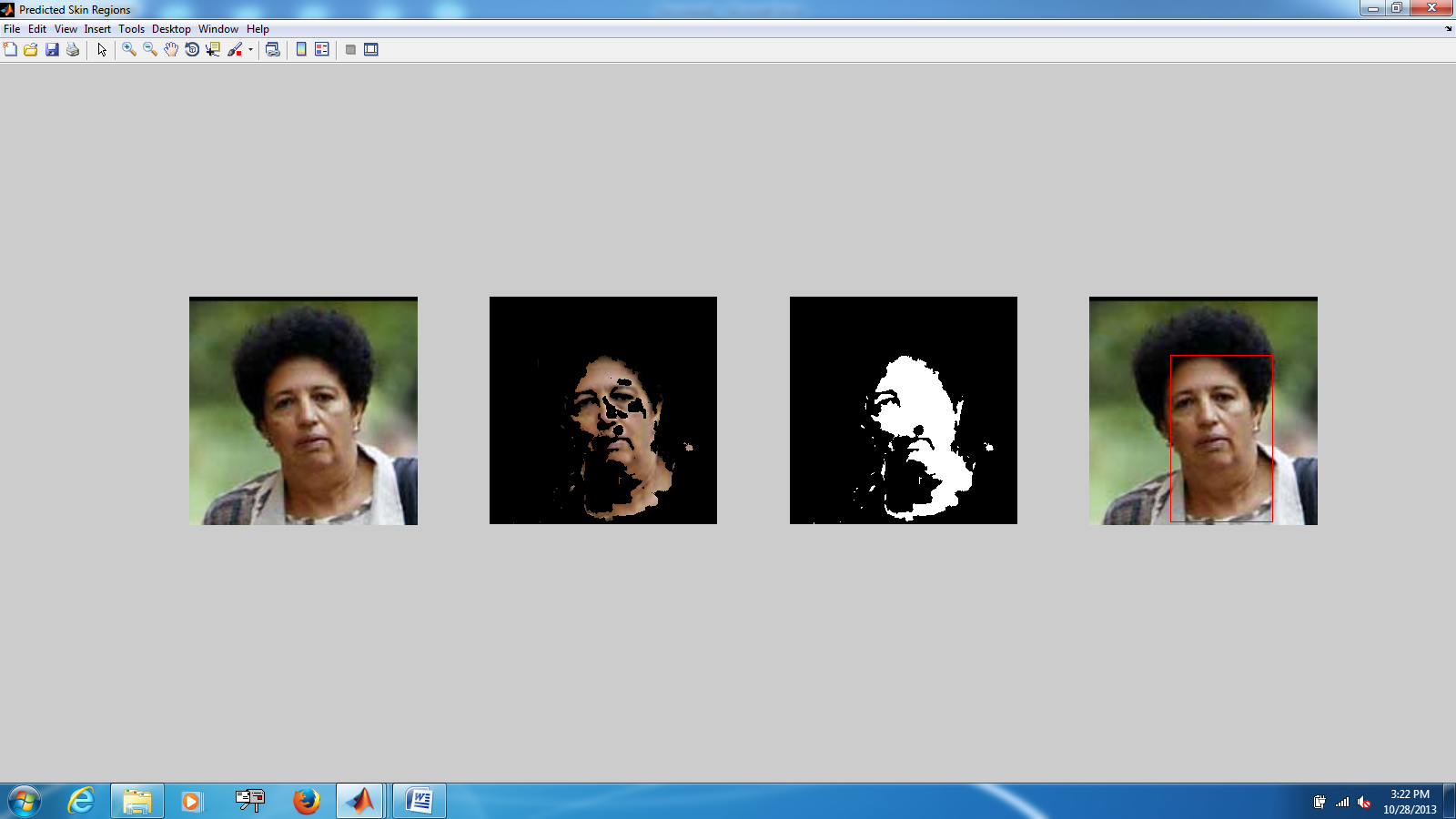
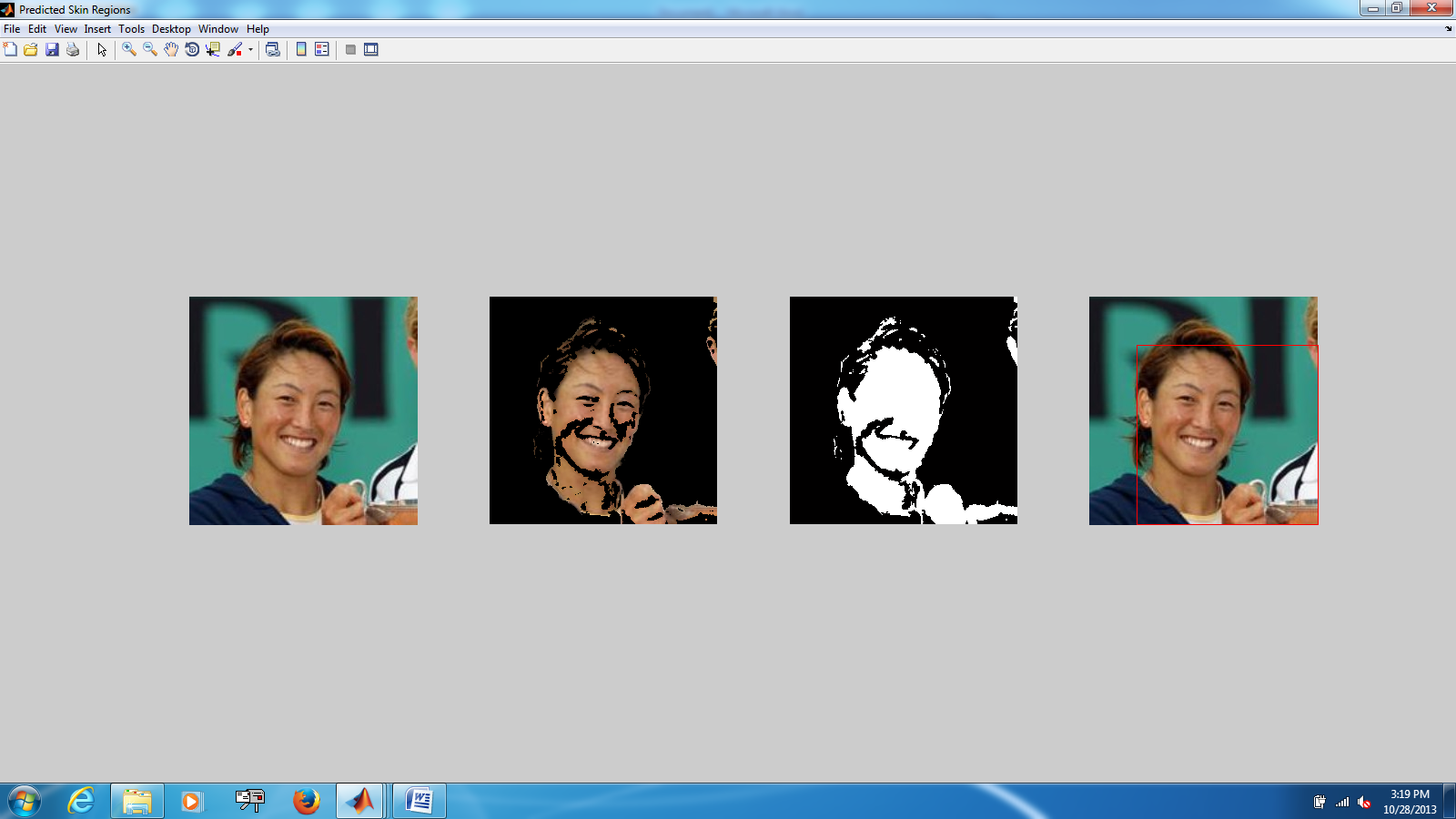
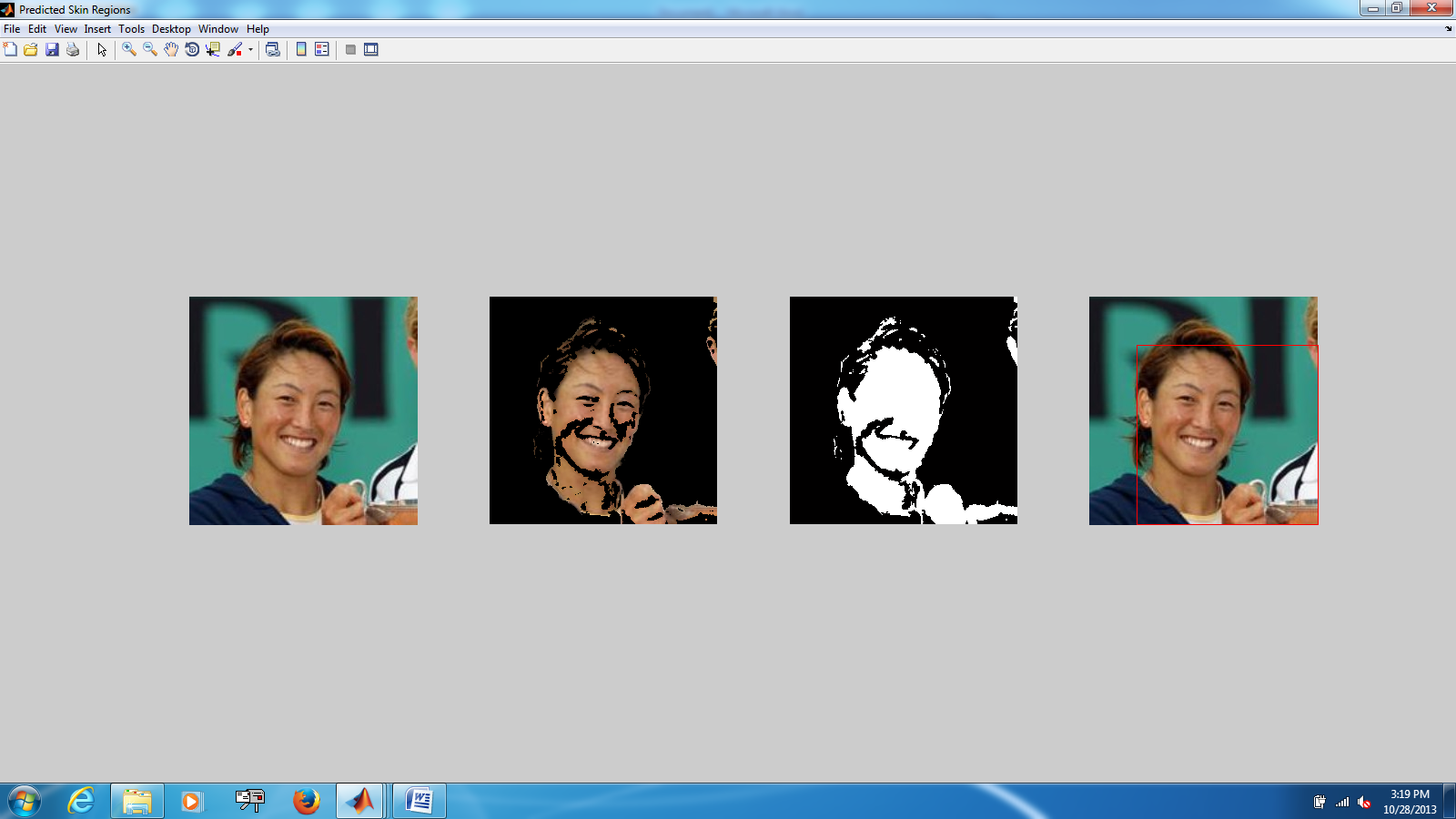
We used the skin model to identify skin pixels in 90 images (30 people and 3 images each). The algorithm completely failed in identifying the skin in 7 of the 90 images; however, all of the failed images suffered presented bad illumination. Furthermore, there were 4 images where the bounding box proposed by the algorithm included a large part of the background. For these cases, the subsequent steps of feature extraction and classification will still work; however, as the images still contain a lot of the background, we may pick up features from the background which we do not want as we want to limit the feature extraction to the face. The Single Gaussian Model is not robust to illumination changes, so we hope to implement a GMM for our final submission. For well-illuminated conditions, the current model is able to pick up the face well (i.e. prominent facial features preserved and bounding box fairly to the face – good separation of face from background) ~95% of the time. The real utility of the Single Gaussian Model is for separating the face from the background in images which present complex backgrounds. The Aberdeen database has fairly homogeneous backgrounds so we do not present any images of our results. Instead, the next section presents the results of testing on images from the Faces in the Wild Database where many images have complex backgrounds.

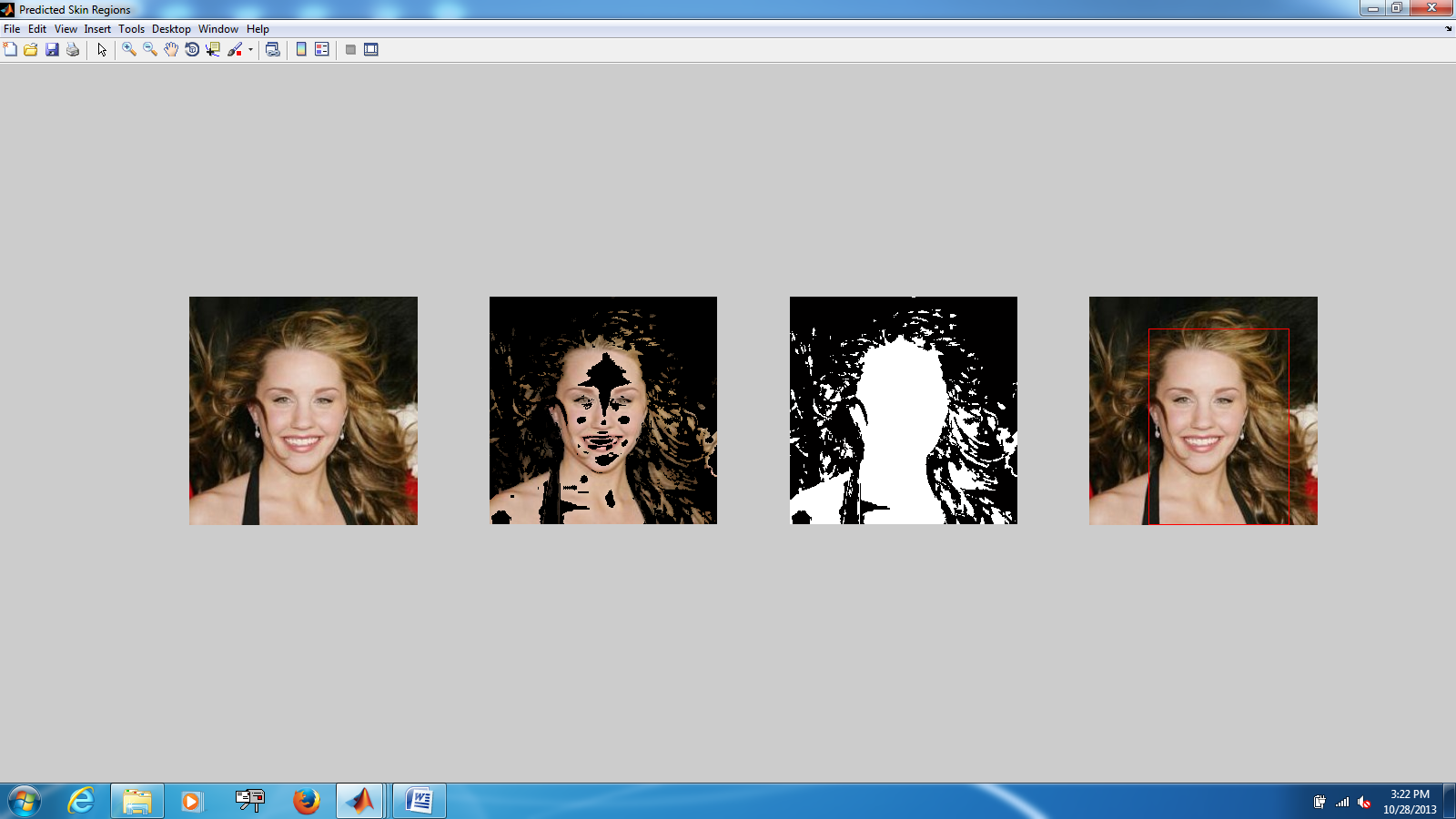
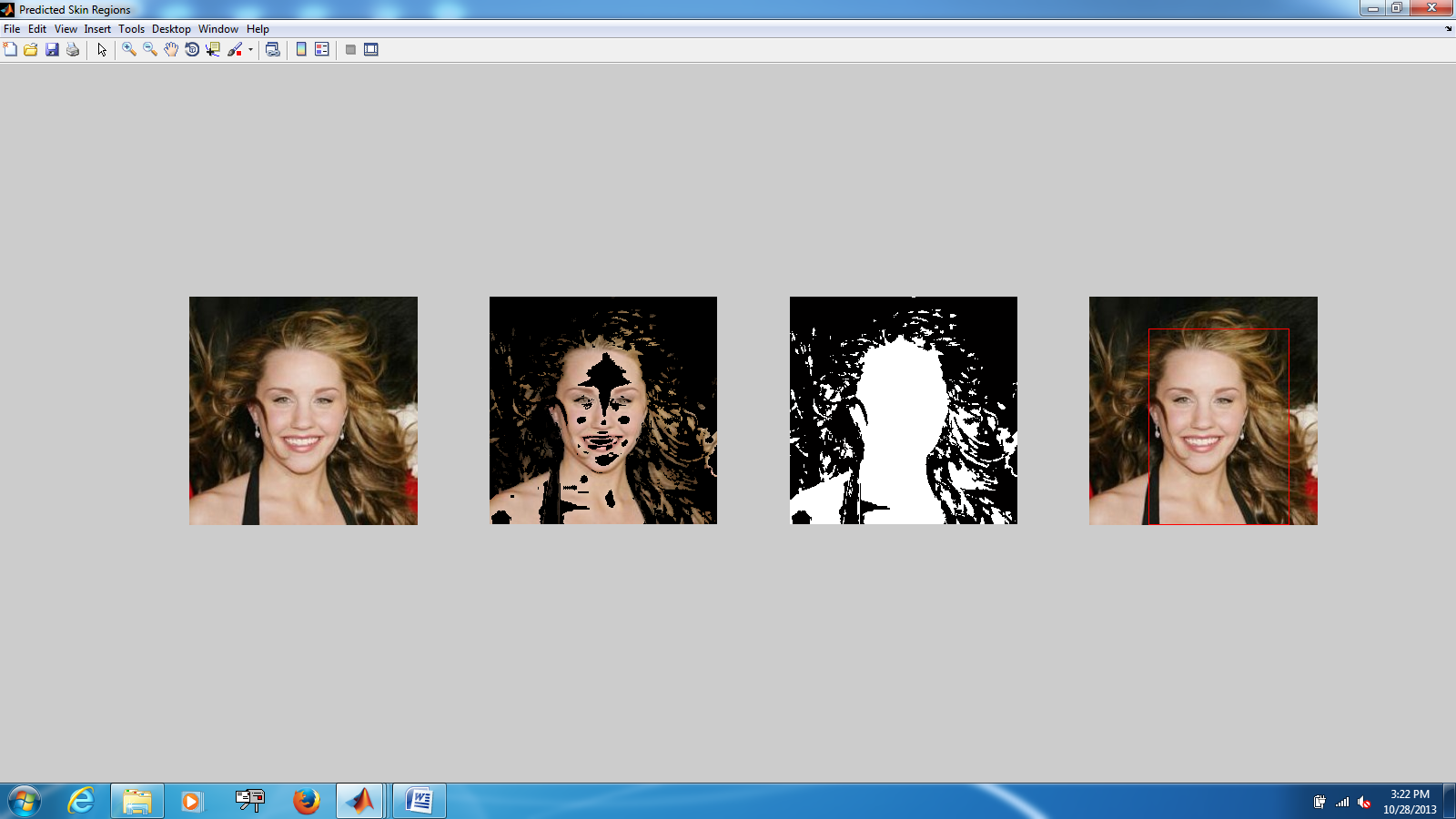
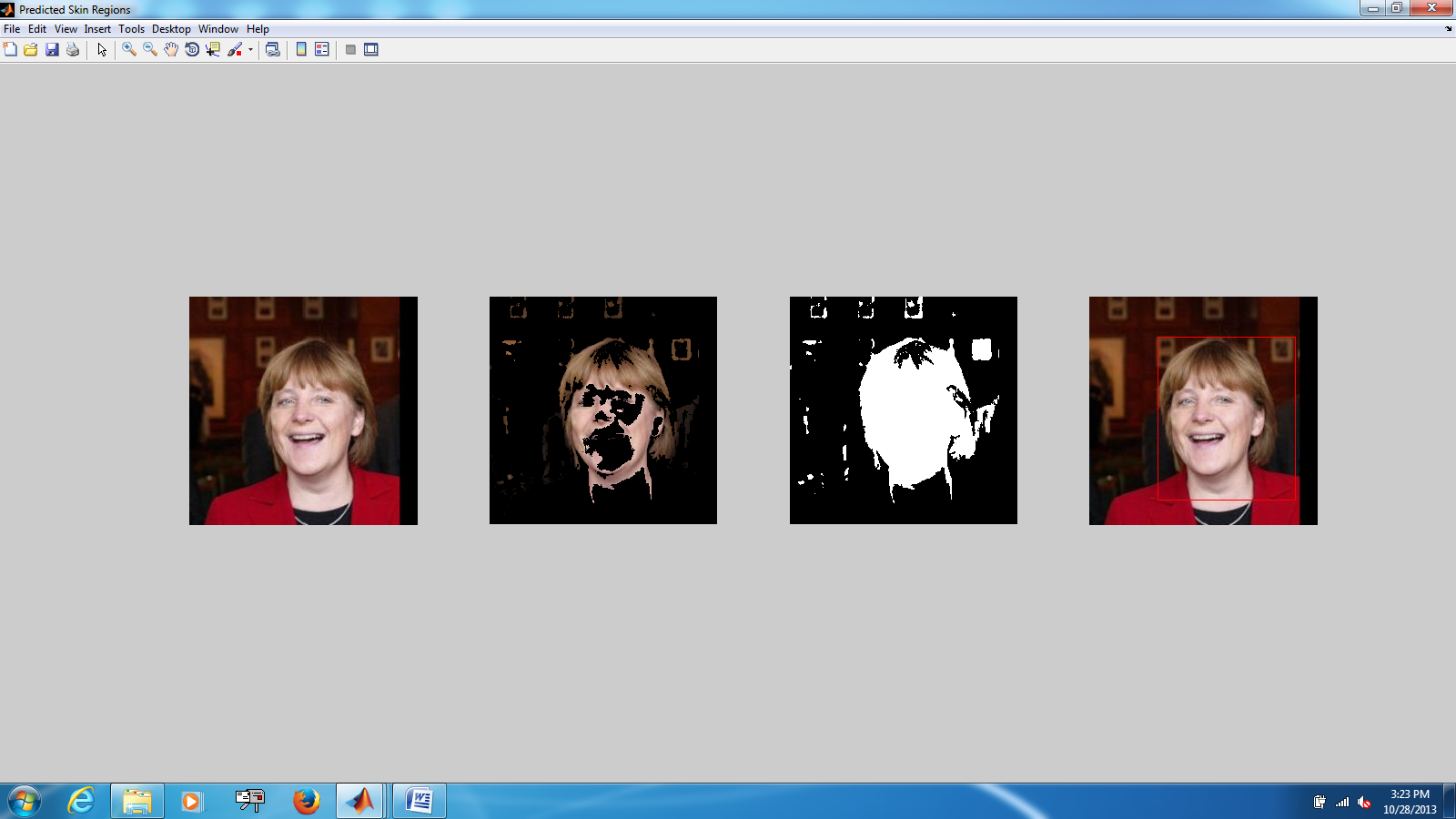
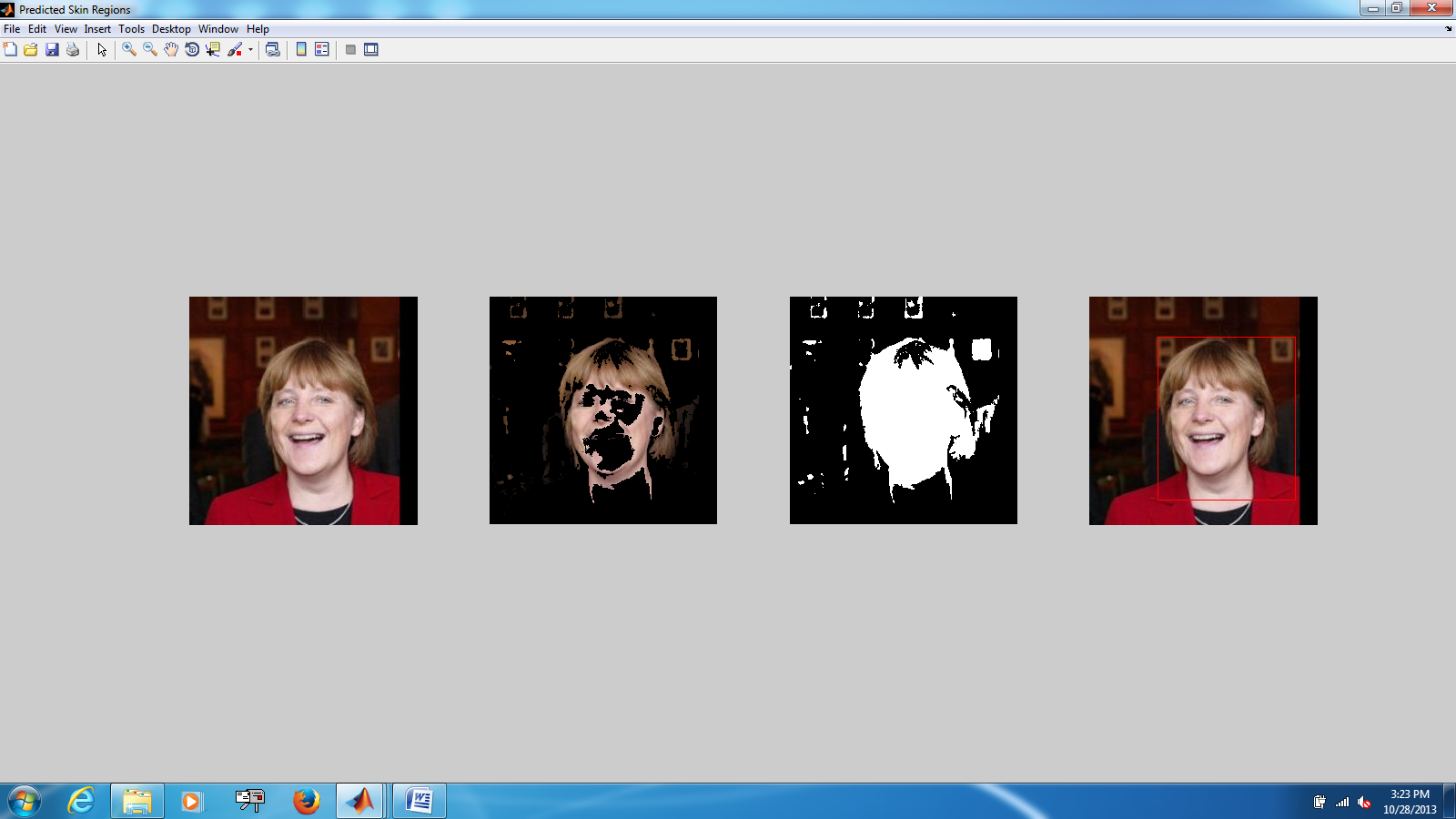
**Testing on the Faces in the Wild Database**

Figure 3 does not show the intermediate steps, but rather shows just the original image and the original image with a bounding box drawn around the identified facial region. The Single Gaussian Skin model appears to work quite well in identifying skin color despite a fair amount of variance in the skin color of the subjects of the input images.

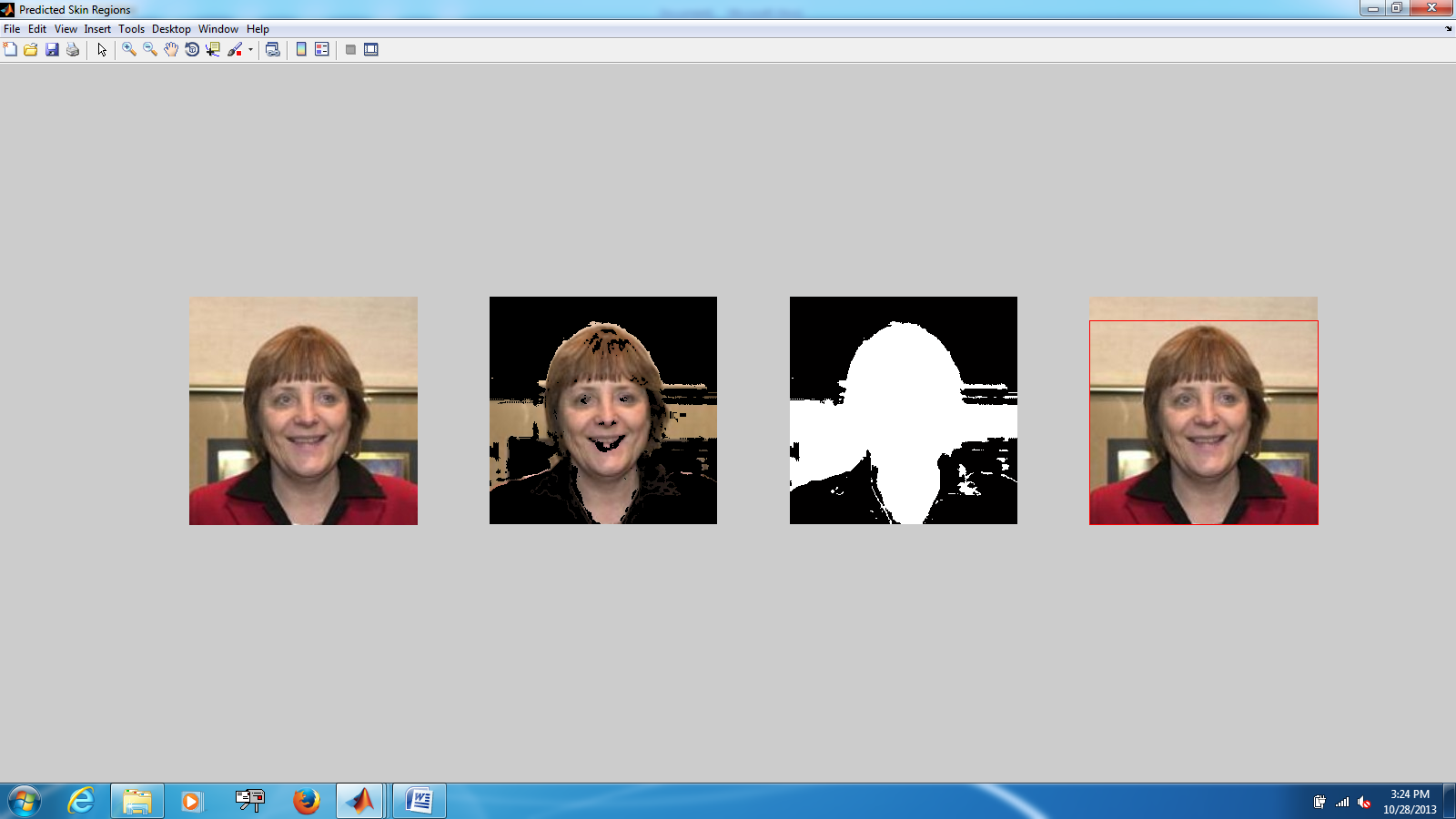
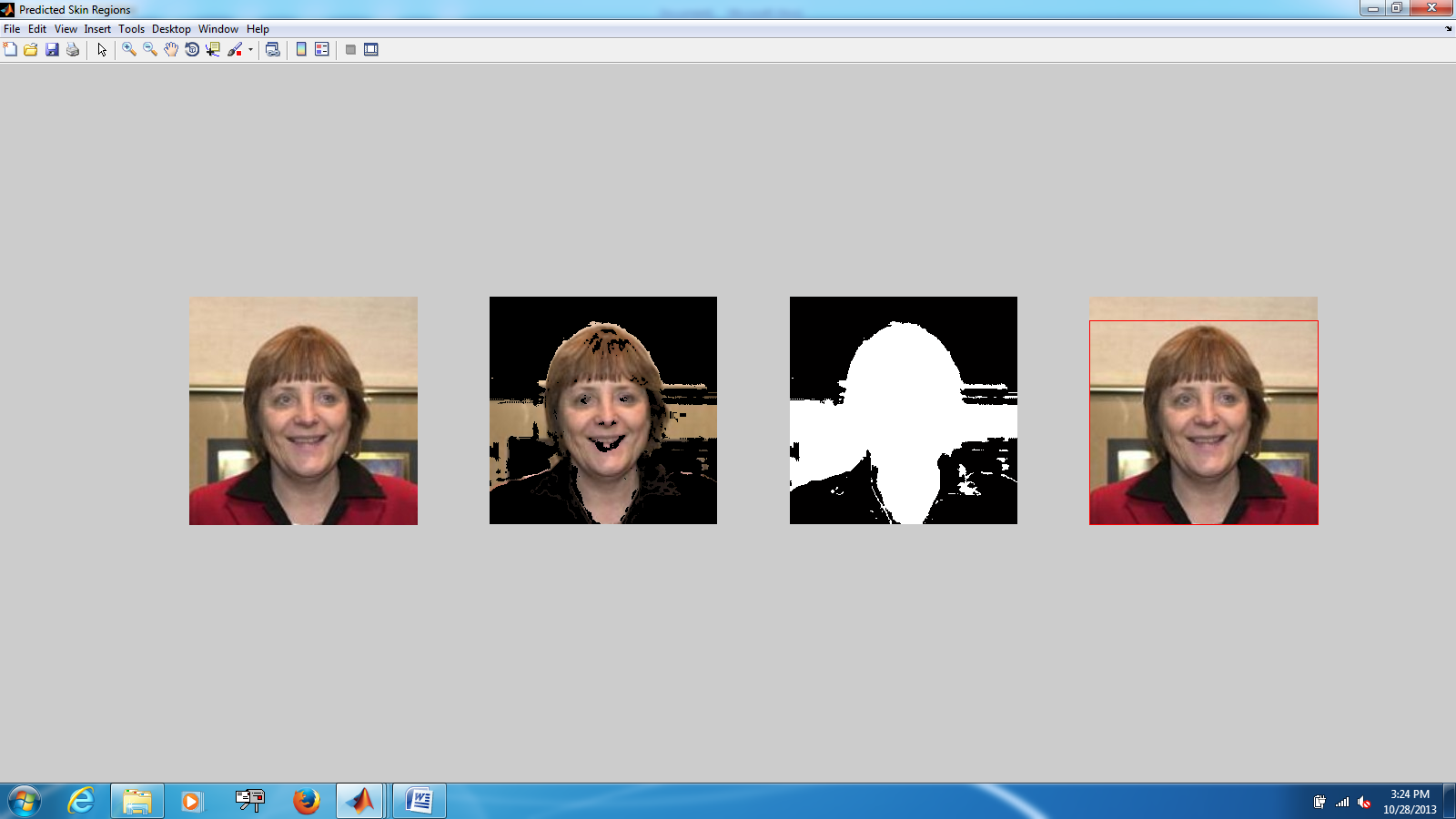
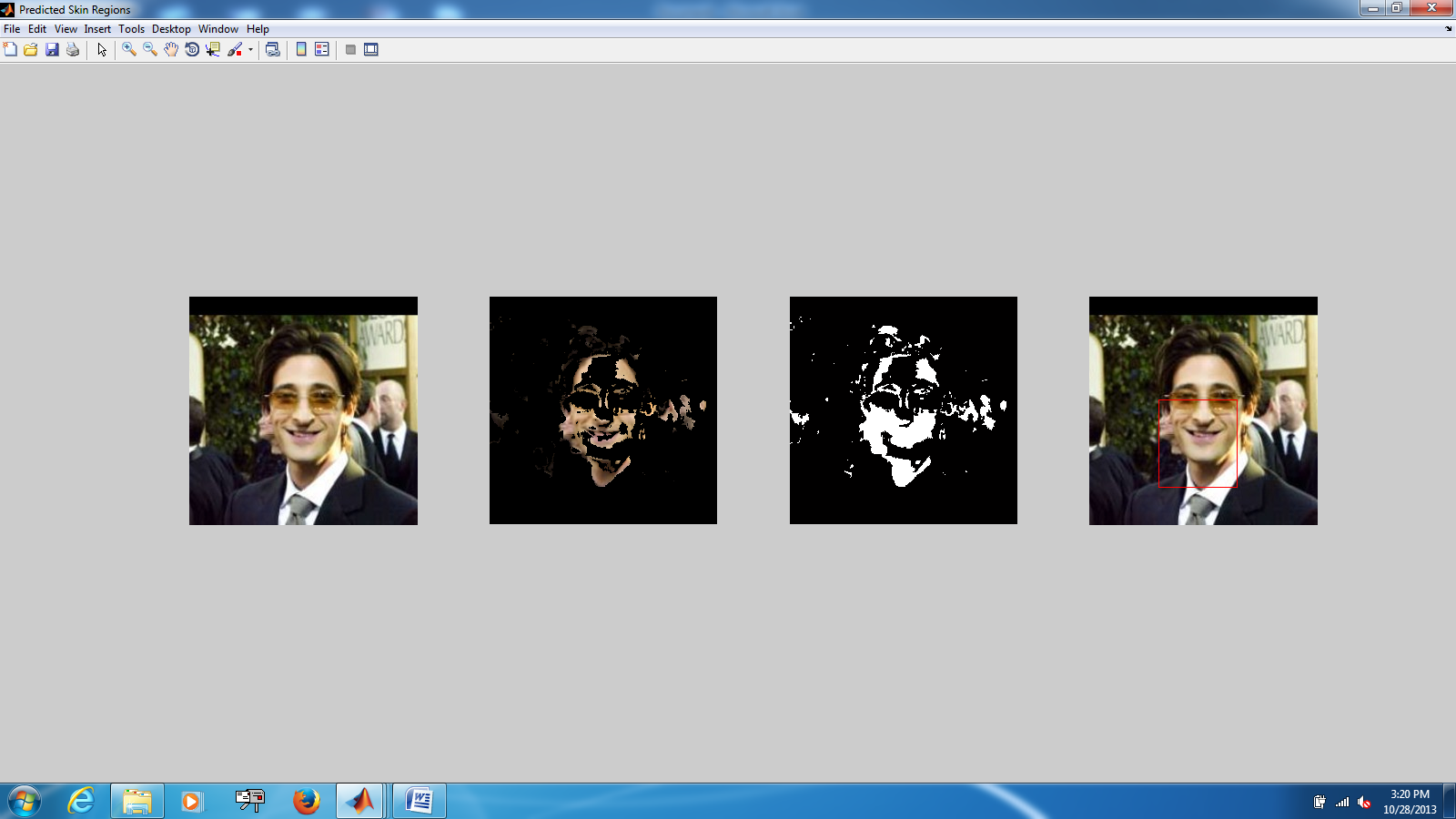
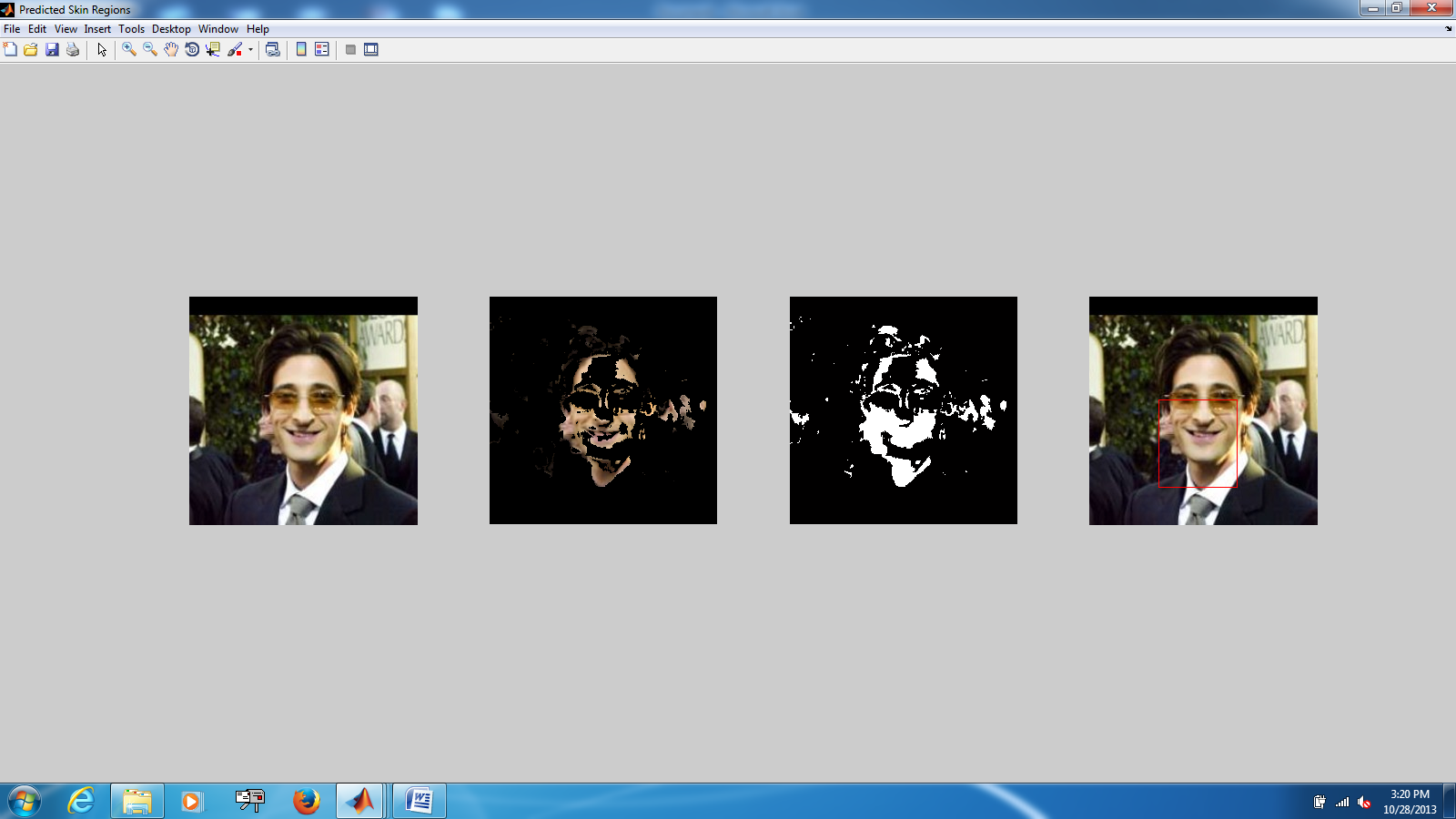
   

Figure 3: (Left) Original images and (Right) resulting bounding boxes for various faces determined from using Gaussian Skin Model for segmentation of faces from complex backgrounds.

We do not make the claim, however, that the Single Gaussian Model *always* works well. In fact, some of the images we have shown (towards the last few images in Figure 3 especially) return bounding boxes that either (1) still include large portions of the background, or (2) do not contain the entire face region. We believe the cause of issue (1) is due to the similarity of the skin color and the background. In this case the background is not very complex, which results in ambiguity between the face/background. We believe the cause of issue (2) has to do with poor illumination as well the presence of an occlusion to the face. The occlusion was significant enough that the face region was not connected during the erosion/filling steps of segmentation and only the largest contiguous region was identified as the face.

We propose to use the Gaussian Mixture Model (GMM) to achieve better results. The GMM is believed to represent varying skin colors and illumination conditions more accurately than the Single Gaussian Model. [7]

## Testing SIFT and the kNN Classifier

We test the feature extraction and kNN classifier stage on the Aberdeen dataset. Our ground-truth database (images we test against) consists of 30 people which have 3 images each. Furthermore, we have a single probe image for each of the 30 person.

First, we use a simple kNN to test a probe image against the database – in this case, for the people in the database, we do not concatenate their feature matrices to form a consolidated matrix. Instead, we compare the feature descriptors of the probe images to feature descriptors of individual images. For 30 test cases, we identify 28 people correctly (93%). If we use a ckNN Classifier described in 2.2.3 for the 30 test images, we are able to identify 29 people correctly (~97%). However, this is a basic dataset which only has some rotational variations and differences in pose. Therefore, SIFT descriptors are able to model the images well.

# Future Work

As alluded to in Section 2, we intend to implement the Gaussian Mixed Model and use this in place of the Single Gaussian Model to improve the representation of varying skin colors in our facial recognition system. We also need to complete our implementation of SIFT and the modifications necessary to fully implement cSIFT in order to address our concerns with how our facial recognition system handles matching in varying illumination conditions.

Successful implementation of these components is the primary goal of our future work leading up to the final presentation; however, we also hope to implement XXXXX in order to reduce the feature correspondences between images based on geometric relationships (i.e. correspondences in the left region of the face match correspondences in other images also in the left region of the face).

Finally, we intend to construct quantitative tests in order to analyze the correctness of the independent components of the project (skin detection/segmentation, feature detection and matching, and identification) as well as the correctness of the facial recognition as a whole.

# Acknowledgment

The members of this project would like to thank Aarati Prasad and the other member of the mHealth Security & Privacy Lab at Dartmouth College for their support in obtaining relevant project data as well as the idea for the project in the first place.

# References

[1] A. Abdel-Hakim, and A. Farag. CSIFT: A SIFT Descriptor with Color Invariant

Characteristics. In *Computer Vision and Pattern Recognition*. 2006.

[2] O. Boiman, E. Shechtman, and M. Irani. In Defense of the Nearest-Neighbor Based

Image Classification. In *Computer Vision and Pattern Recognition*. 2008.

[3] D. Lowe. Distinctive image features from scale-invariant keypoints.

*International Journal of Computer Vision*. 60(2) 2004, pp. 91-110.

[4] A. Vedaldi, 2006. An implementation of SIFT detector and descriptor.

*Unpublished - UCLA.* <http://www.robots.ox.ac.uk/~vedaldi/assets/sift/sift.pdf>

[5] Labeled Faces in the Wild database: <http://vis-www.cs.umass.edu/lfw/>

[6] Aberdeen: <http://pics.psych.stir.ac.uk/2D_face_sets.htm>

[7] T.S. Caetano, S.D. Olabarriaga, D.A.C. Barone, Performance evaluation of single and multiple-Gaussian models for skin-color modeling, SIBGRAPI02, 2002.

[8] A.E. Abdel-Hakim and A.A. Farag, CSIFT: A SIFT Descriptor with Color Invariant Characteristics, *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 1978-1983, 2006.