Proceedings of the 10th IASTED International Conference
SIGNAL AND IMAGE PROCESSING (SIP 2008)
August 18-20, 2008 Kailua-Kona, HI, USA

FACE AND LIP LOCALIZATION IN UNCONSTRAINED IMAGERY

Brandon Crow and Jane Xiaozheng Zhang
Department of Electrical Engineering
California Polytechnic State University
1. Grand Ave. San Luis Obispo, CA 93401
USA
{bcrow, jzhang}@calpoly.edu

ABSTRACT

When combined with acoustical speech information, visual speech information (lip movement) significantly improves Automatic Speech Recognition (ASR) in acoustically noisy environments. Previous research has demonstrated that visual modality is a viable tool for identifying speech. However, the visual information has yet to become utilized in mainstream ASR systems due to the difficulty in accurately tracking lips in real-world conditions. This paper presents our current progress in addressing this issue. We derive several algorithms based on a modified HSI color space to successfully locate the face, eyes, and lips. These algorithms are then tested over imagery collected in visually challenging environments.

KEY WORDS

Automatic visual speech recognition, face tracking, lip tracking, target localization, mean-shift, Bhattacharyya coefficient

1. Introduction

While mouse keypad interfaces have dominated as a main human-computer interface for more than two decades, voice user interfaces are beginning to emerge as a promising way to communicate between humans and computers and can profoundly change the way we live. It is noted that “browsing the web with a five-line screen and a tinny number pad is not a very gratifying experience” [1]. The use of voice interface on the other hand is more natural, easier, and safer to use. For example, one can compose an email message, or open a program using speech without touching the keyboard. Voice interface is especially promising when the users’ eyes and hands are busy or the users are mobile [2], such as in a car environment. A driver can select a radio station, or operate an air conditioner by talking to the dashboard avoiding any distractions of the eyes and hands. Voice interface can also be very helpful in assisting people who can not see, or can not point and click and type, such as computer users with repetitive strain injuries.

The key technology that permits the realization of a pervasive voice user interface is automatic speech recognition (ASR). Since its invention in the 1950s, ASR has witnessed considerable research activities and in recent years is finding its way into practical applications as evidenced by more and more consumer devices such as PDAs and mobile phones adding ASR features. While mainstream ASR has focused almost exclusively on the acoustic signal, the performance of these systems degrades considerably in the real-world in the presence of noise. One way to overcome this limitation is to supplement the acoustic speech with a visual signal that remains unaffected in noisy environment. This idea is motivated by humans’ ability to lipread. As the cost of cameras continues to drop, incorporation of visual modality is becoming a feasible solution in many practical applications.

While previous research [3, 4] demonstrated that visual modality is a viable tool for identifying speech, the visual component has yet to become utilized in mainstream ASR system. To-date, in almost all research in audio-visual ASR, work has concentrated on visually clean data – data collected in a studio-like environment with controlled lighting conditions, uniform background and high video quality. There is a high demand for creating a robust visual front end in realistic environments. For example, one of the most compelling places to embed speech is in the car. An “in-vehicle” voice response system promises safer driving through “hands-free, eyes-free” operation of the cell phone, dashboard controls and navigation system. Because of environmental conditions such as road noise and wind noise, acoustic-only speech recognition in a moving automobile is a very hard problem. In this situation, supplementing the acoustic channel with visual speech information can have extreme potentials in improving speech recognition performance.

Building on our previous work [5, 6], this paper presents our current progress on developing a robust visual front-end that allows accurate localization of face and lips in a real-world environment. This represents one of the key steps toward attaining a successful audio-visual speech recognition solution.

2. Proposed System

Figure 1 depicts the top-level design of our proposed system. While our system is currently a work in progress, several modules have been completed and the *Extract Face Coordinates* and *Extract Lip Coordinates* modules will be detailed in the following subsections. A brief description of the system's functionality follows.

The system module is responsible for extracting lip parameters for future downstream audio speech integration. However, lips are not easily detectable due to their relative size and ever-changing shape. To simplify localization of the lips we will first locate the face.

Previous research has shown a modified HSI color space is robust in skin localization [5]. The modification consists of shifting the hue component to the right by +0.2, assuming a 0 to 1 (double) representation of the hue. Shifting allows the face/lip hue color to fall within a connected region, leading to a simple model describing the skin hue variation using the Gaussian distributions with an ensemble average of 0.14 (0.34 after shifting) and a standard deviation of 0.11 [6]. Therefore, all image processing within the system module will use the modified HSI color space.

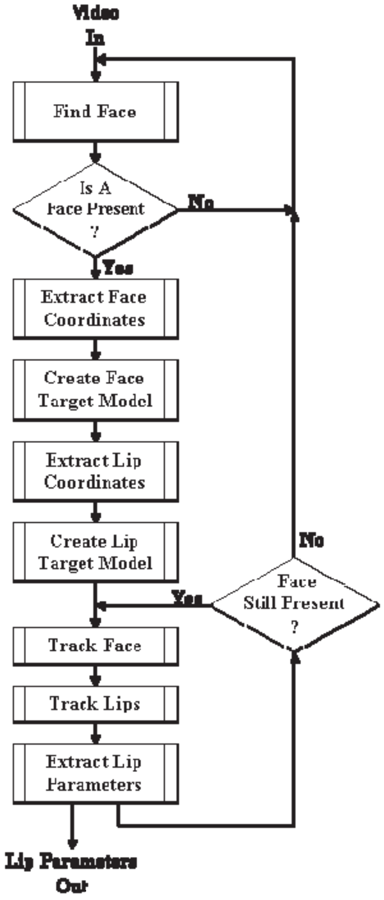

Figure 1. Top-Level Design of Proposed System

The *Find Face* module is based on the work of [8]. The Bhattacharyya coefficient is used as it provides both scale and rotational invariance in comparing a potential face (target) to a face model. Our method, however, uses 16x16 HS bins, or 1/64 of the 128x128 nRG bins implemented in [8]. This drastic reduction in memory and processing requirements are essential to ensure our final system will run at a minimum of real-time – 30 frames per second (fps).

To track both the face and lips (the *Track Face* and *Track Lips* modules of Figure 1), the mean-shift algorithm is used with an Epanechnikov kernel, similar to [9].

2.1 Extract Face Coordinates Module

Within the *Find Face* module five elliptical regions-of-interest (ROIs) are selected from the current video frame and compared with three face models (light, medium and dark skin tone) via calculation of the Bhattacharyya coefficient. This is accomplished using the modified hue and saturation color spaces and equations adapted from [8]. The five ROIs are arranged similar to [8] but their individual sizes are based on the size of the frame and the height/width ratio of 1.2 (average ratio for a face). However, to avoid the detection of a face that is partially occluded by the edge of a frame, perimeter regions (based on frame size) are excluded from ROI placement. This can alter the effective height/width ratio. For example, a 360x240 frame size produces ROI sizes of 69x90; resulting in a height/width ratio of 1.3. The three face models are created by manually selecting the face (excluding head-hair and neck) of three subjects. Each model is then converted to their representative pdf forms and stored offline (prior to invoking the system). The three face models can be seen in Figure 2.

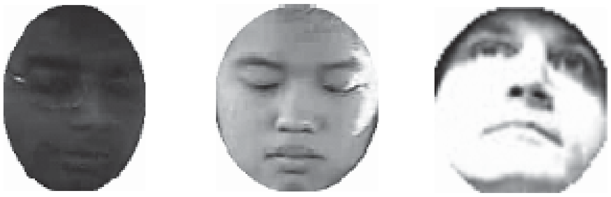

Figure 2. Three Face Models

Masking the current frame with the five ROIs generated, each ROI is converted to its representative pdf for comparison to the three models. The Bhattacharyya coefficient is then calculated for each ROI and model, if the maximum coefficient value is greater than the experimentally determined threshold of 0.6, the system concludes a face is present in the frame and this ROI is passed to the *Extract Face Coordinates* module.

While a calculated Bhattacharyya coefficient can determine if a face is present within a given ROI, it does not guarantee the complete face is within the ROI; this is illustrated in Figure 3, where the top-right ROI is detected as a face even though it contains only a portion of it. Therefore, the *Extract Face Coordinates* module is responsible for adjusting the ROI found in the *Find Face* module to that of the complete face. This is accomplished using statistically determined thresholds and simple binary morphology.

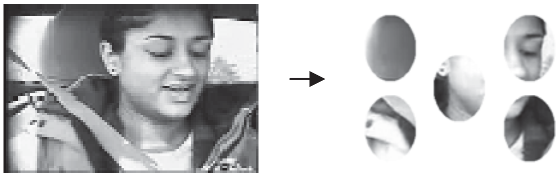

Figure 3. Example of Five ROI Frame Selection from the Find Face Module (displayed in grayscale for clarity)

While saturation is invaluable in locating features in a region, it is a poor metric for locating a feature rich region such as a face since saturation values can vary drastically. However, with the exception of shadows, intensity values remain relatively constant for a given region. Additionally, using the modified HSI color space, skin hue falls between the values of 0.01 and 0.67. Therefore, both hue and intensity are chosen as metrics for video frame feature discrimination.

It is assumed that a face detected within a given frame is illuminated from a source located in front of the facial plane (i.e. not back-lit) and, therefore, a minimal value of 0.2 in intensity is used as a threshold to remove dark/shadowed regions of the passed ROI. The median and minimum value of the ROI's resulting intensity space is then calculated to determine the frame's intensity threshold:

$$I_{th} \quad \frac{median[\,I\,]-min[\,I\,]}{2} \tag{1}$$

where I is the intensity space and the median and minimum are calculated using only the non-zero values of intensity.

Since the ROI's Bhattacharyya coefficient is at least 0.6, we can assume a predominant portion of the region is skin. Therefore, calculating the median intensity value provides an expected skin intensity value, removing outliers of dark regions and background. Additionally, since our experiments have determined that facial intensity values are Gaussian distributions, subtracting the minimum intensity value provides an approximation to the 3 standard deviation envelope. In other words, I_{th} is a discriminatory metric, biased toward skin statistics based on the assumption of a Gaussian distribution.

As previously mentioned, using a modified hue space, skin hue falls primarily between the values of 0.01 and 0.67. However, a pixel with a hue value greater than approximately 0.45 in the modified HSI color space has a higher probability of belonging to a non-skin pixel[6]. Therefore, an upper threshold of 0.4 is implemented within the ROI's modified hue space and the median and standard deviation for the modified hue space is then calculated, excluding values equal to 0.

The statistical values calculated within the ROI for both intensity and hue space are then used as thresholds across the entire frame's hue and intensity space to produce a binarized image. To remove noise and erroneous connected components, the binarized image is then eroded using a 3x3 mask.

The largest blob remaining in the frame, and nearest the *Find Face* module's selected ROI, is the face. Summation and smoothing across the rows results in a column signal where the edges of the maximum region correspond to the left and right of the face. Using these coordinates, as well as a face height/width ratio of 1.2, a bounding box is created and 'slid' down the rows to locate maximum pixel containment. Requiring the summation of pixel containment within the bounding box to exceed any previous row's summation by 2% biases the "slide" towards the top of the image, removing potential coordinate selection errors based on neck visibility. In other words, while the typical face has a height/width ratio of approximately 1.2 and since this module selects skin, the neck of an individual would affect the extracted bounds. Examples of this processing can be seen in Figure 4.

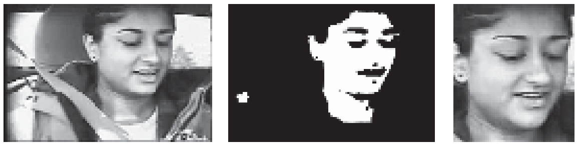

Figure 4. Example of Extract Face Coordinates Module

The extracted bounds are then oversized to counteract erosion effects as well as add background pixels. The addition of background pixels is implemented to minimize the number of iterations the mean-shift algorithm will perform downstream. These coordinates are then passed to both the *Create Face Target Model* and *Extract Lip Coordinates* modules.

2.2 Extract □ip Coordinates Module

This module is similar to the *Extract Face Coordinates* module but uses saturation and intensity rather than hue and intensity. Intensity is chosen as a metric due to the low values at the mouth's opening regardless of it being open or closed. Additionally, saturation is a robust feature discriminator and edge detector, making it ideal for localizing facial features and their parameters.

The ROI received by this module contains only a face and a perimeter consisting of background noise. Therefore, selecting a sub-region, centered about the ROI, assures statistical analysis will only take into account the face. Using the statistical measures of median and standard deviation for saturation and intensity, several thresholds can be implemented to select and remove the skin of the face from the ROI, leaving only the facial features within the frame.

The median values for saturation and intensity are used in lieu of the mean to remove the pixels associated with darker features within the sub-region; such as lips, eyebrows, and other facial hair. This exclusion, therefore, approximates the means of the skin's saturation and intensity.

The standard deviations for the saturation and intensity space are calculated separately for both the rows and columns of the sub-region. The minimum standard deviation between the rows and columns for both saturation and intensity is selected to better approximate the skin's standard deviations, since these calculations will also include facial features.

Since very low saturation values are gray-tone, ranging from white to black, removing all pixels associated with saturation values below 3 standard deviations from the median will remove noise associated with regions less likely to be skin. Additionally, removing pixels with an intensity value lower than the experimentally determined 1.5 standard deviations from intensities median will remove facial regions less likely to be skin (i.e. eyes, nostrils, mouth opening, and facial hair). Specifically:

$$S_{th} \quad median[\,S\,] - 3\cdot std[\,S\,] \tag{2}$$

$$I_{th} \quad median[\,I\,] - 1.5\cdot std[\,I\,] \tag{3}$$

where S and I are saturation and intensity spaces, respectively.

A binarized image, *SI*, is then constructed from the resulting saturation and intensity spaces; where a pixel is 1 only if both the corresponding saturation and intensity pixel is non-zero. Therefore, skin now has pixel values of 1, while facial features are 0; this is illustrated in Figure 5.

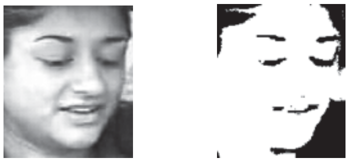

Figure 5. Example of Binarized image *SI*

Let R equal the summation and smoothing across ~*SI*'s (inverted *SI*) columns. Differentiating this spatial signal, dR, and determining the maximum value of dR for the upper 1/3rd provides an approximate row index for the top of the eyes, R_E.

The left and right side of the face can be determined by summing *SI* across its rows, resulting in a spatial signal, C, that can be used to determine the left, C_L, and right, C_R, edges of the face, as follows:

$$C_M \quad max[\,C\,] \tag{4}$$

$$C_L \quad min[\,C(\,C_M\,,1)] \tag{5}$$

$$C_R \quad min[\,C(\,C_M\,,length(\,C\,))] \tag{6}$$

Note that C_L is the minimum value found in the decreasing direction of C from C_M. This ensures C_L is the first minimum nearest the face (closest edge).

These column coordinates are used to generate a generic face mask, which has a predefined shape but its height and width is determined by the width of the face (column coordinates) and the facial height/width ratio of 1.2. It is then placed using the three facial coordinates – C_L, C_R, and R_E. This masking step is used to remove large portions of non-facial feature regions (noise) and is applied to ~*SI*. Finally, this binary result is multiplied by *S*, resulting in an image, *F*, comprised primarily of facial features. An example of the mask and *F* can be seen in Figure 6.

However, noise may be prevalent below the chin as well as along the sides of the face due to shadowing. Therefore, the lower lip region, R_{LL}, is located based on face width and eye location, R_E, using the following equation:

$$\begin{aligned} &R_{LL} \quad min[\,dR(r_1,r_2)] \\ &r_1 \quad R_E + 0.6\cdot(C_R - C_L)= \\ &r_2 \quad R_E + 0.75\cdot(C_R - C_L) \end{aligned} \tag{7}$$

All pixels below R_{LL} are then set to zero, resulting in the updated image *F*, far right image of Figure 6.

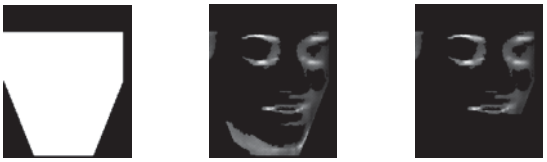

Figure 6. Example of Mask, *F* and Updated *F*

To improve both the accuracy of the lip coordinates (additional noise removal), and the mouth parameters (opening width and height), the face's angle from the horizontal is calculated. This is accomplished by calculating the width of the eye span, C_{EW}, and the top of the left, R_{EL}, and right, R_{ER}, eye regions. Since the eye region tops are located near the center of each eye and the eye span width is approximately 1.5 times that of the eye centers, the following equation is used to calculate the facial tilt, α_F:

$$\alpha_F \quad arctan\left(\frac{3}{2}\cdot\frac{R_{ER}-R_{EL}}{C_{EW}}\right) \tag{8}$$

To locate the top and bottom of the mouth an upper bound for searching R is determined as

$$R_{UL} \quad R_E + C_{EW}\cdot(1+sin(-\alpha_F))= \tag{9}$$

The upper, R_U, and lower, R_L, mouth opening is then determined by using (10), then (11) and (12).

$$R_M \quad max[\,dR(\,R_{LL}\,,R_{UL}\,)] \tag{10}$$

$$R_U \quad ZC[\,dR(\,R_M\,,R_{LL}\,)] \tag{11}$$

$$R_L \quad ZC[\,dR(\,R_M\,,R_{UL}\,)] \tag{12}$$

where ZC is the zero-crossing point.

The left and right edges of the mouth are determined by summing and smoothing F across R_U and R_L, generating a spatial signal of the columns about the lip region's rows, C'. Replacing C with C', (4) then (5) and (6) are used to determine the left edge, C'_L, and right edge, C'_R, of the lips. Noise may still be present in C' due to shadows, facial blemishes, etc. Therefore, Let C_{LW} be the calculated width of the lips:

$$C_{LW} \quad C'_R - C'_L = \tag{13}$$

then if

$$C_{LW} \geq \frac{2}{3} C_{EW} \tag{14}$$

$_{LW}$ includes noise. Since the lip edge pixels, on average, have larger values than the mean of C',

$$C' \quad C' - mean[\,C'\,] \tag{15}$$

Analysis of equations (4), then (5) and (6) is then repeated to determine C'_L and C'_R. The lip region is then defined by the coordinates C'_L, C'_R, R_U and R_L (left, right, upper, lower). An example of the selected region can be seen in the right image of Figure 7. This example invoked equation (15) due to the shading on the right side of the lips (top-left image of Figure 7). The effects of equation (15) can be seen in the bottom-left image of Figure 7.

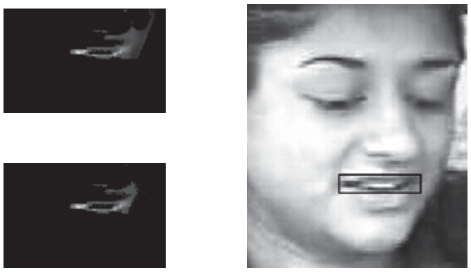

Figure 7.Example of Lip Coordinates

□. Results

All testing was performed using MATLAB R2006a on a desktop PC with a 2.93 GHz Celeron processor and 1.0 GB of memory. A total of 7 video files are used for testing. The first frame in which a face is detected is then passed to the *Extract Face Coordinates* and *Extract Lip Coordinates* modules. 5 of the videos are from [7]. The remaining 2 were created using a Channel Plus 7120 ¼" CCD camera with the same subject but different clothing; including a beanie and long-sleeve shirt of similar hue to that of skin in an effort to challenge the system.

The average runtime for the *Extract Face Coordinates* module is 278.4 ms/frame (~4 fps). While this is a slow runtime for a system designed to run at real-time, once the system is finalized it will be coded in C, providing at least an order of magnitude improvement in runtime. Additionally, the *Extract Face Coordinates* module is only invoked if a face has not been previously located; otherwise the mean-shift algorithm (via the *Track Face* module) tracks the face. Finally, the average runtime for the *Extract Lip Coordinates* module is 43.2 ms/frame (~23 fps).

Success for the *Extract Face Coordinates* module is defined as full inclusion of facial features with the face centered within the region and a minimum of 60% of the region's pixels comprised of the face (i.e. 40% background, neck and hair).

Success for the *Extract Lip Coordinates* module is defined as full inclusion of inner lip mouth opening with the four coordinates – C'_L, C'_R, R_U and R_L, within 3 pixels of their true position.

Test results for the *Extract Face Coordinates* module is 100% (7/7); these results can be seen in the center column of Figure 8. Test results for the *Extract Lip Coordinates* module is 86% (6/7), and can be seen in the right column of Figure 8. The single failure was a 4 pixel error in the upper lip coordinate, R_U, shown in the top-right of Figure 8.

□. Conclusion

In this paper we presented two modules of our lip parameter extraction system. Based on five regions of interest and their respective Bhattacharyya coefficients, the approximate location of a face can be determined. Our modules then accurately locate the face and lips for downstream processing.

Current work is focused on increasing the accuracy of the *Extract Lip Coordinates* module, enabling removal of the *Track Lips* and *Create Lip Target Model* modules. In the future physical dimensions of the lips will be extracted based on the identified lip region and input to a recognition engine to perform automatic speech recognition.

□c□no□led□ement

This work was sponsored by the *Department of the Navy, Office of Naval Research*, under Award # N00014-04-1-0436.

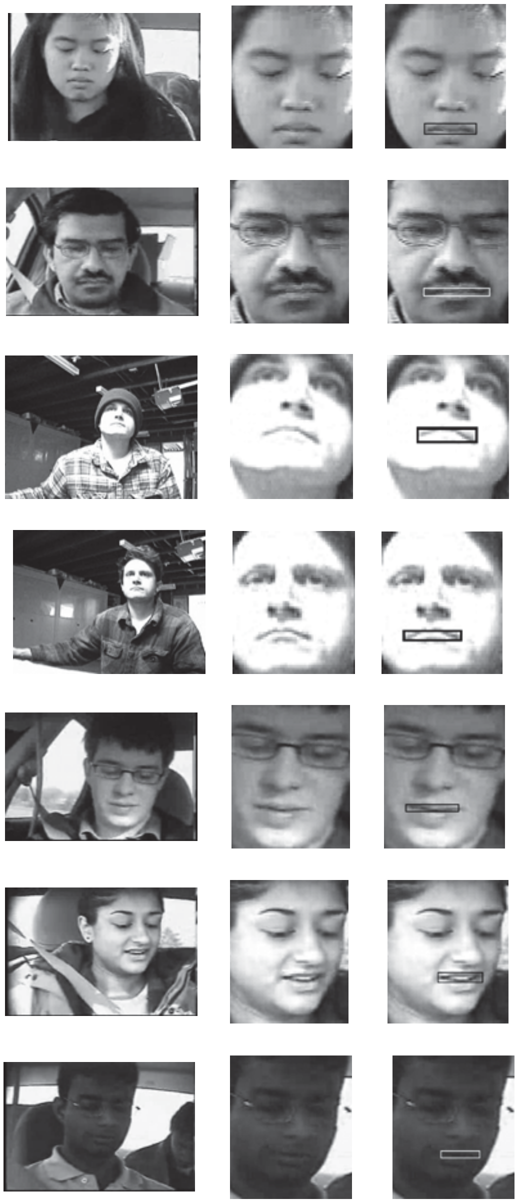

Figure 8.Test Results for *Extract Face Coordinates* and *Extract Lip Coordinates* modules

□e□erences

[1] Katie Hafner, The Future of Cellphones Is Here. Sort Of. *New York Times*, Feb. 2002.

[2] M.G. Helander, T.K. Landauer, & P.V. Prabhu (Eds.), *Handbook of human-computer interaction* (New York, NY: Elseview Science Inc., 1997).

[3] D.G. Stork & M.E. Hennecke, *Speechreading by Humans and Machines* (Berlin, Germany: Springer-Verlag, 1996).

[4] G. Potamianos, et al., Audio-Visual Automatic Speech Recognition: An Overview, *Issues in Visual and Audio-Visual Speech Processing* by G. Bailly, E. Vatikiotis, and Perrier, Eds. (Cambridge, MA: MIT Press, 2004).

[5] X. Zhang, H.A. Montoya, Statistical Modeling of Lip Color Features in Unconstrained Imagery. *Proc. 11th World Multiconference on Systemics, Cybernetics and Informatic*, Orlando, Fl, 2007.

[6] X. Zhang, H.A. Montoya, & B. Crow, Finding Lips in Unconstrained Imagery for Improved Automatic Speech Recognition. *Proc. 9th International Conference on Visual Information Systems*, Shanghai, China, 2007, 185-192.

[7] B. Lee, M. Hasegawa-Johnson, C. Goudeseune, S. Kamdar, S. Borys, M. Liu, & T. Huang, AVICAR: Audio-Visual Speech Corpus in a Car Environment. *INTERSPEECH-ICSLP*, 2004.

[8] D. Comaniciu & V. Ramesh, Robust Detection and Tracking of Human Faces with an Active Camera. *Proc. Third IEEE International Workshop on Visual Surveillance*, Dublin, Ireland, 2000, 11 – 18.

[9] D. Comaniciu & V. Ramesh, P. Meer, Kernel-Based Object Tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *25*(5), 2003, 564 – 577.