Abstract—In this paper, I present a method for determining the location of human hands from a video recording of wide gestures. This method utilizes various techniques in image processing, including color filtering, motion filtering, noise reduction, and region discovery. Additionally, it proposes the use of optical flow as additional information in determining the rough location of the hands.

I. INTRODUCTION

In the study of computer vision, much work has been done on human subject detection, and to a lesser degree, gesture detection. Gesture detection is the process of, from an image or video, programmatically determining the speed, location, direction, and/or form of various components of the body in order to extract some symbolic meaning. This task is complicated by both the variation amongst different subjects at different scales, in different lighting, and in different clothing, and, more severely, by the range of articulated motion in the body, which may include self-occlusion. Some efforts have been made to compensate for these variations. [9] utilizes a color-based detector in conjunction with a trained hand detector. And [6] has utilized stereovision to improve localization of the hands. These have proven to be quite effective methods.

For the purposes of development on mobile (i.e., Android and iPhone) platforms, however, it becomes substantially more desirable, if not necessary, to minimize computational costs and eliminate dependency on secondary resources such as stereovision. To that end, I propose a simple five-component method of hand localization as it pertains to wide gesture recognition. This method uses one stage of pre-processing, one stage of color filtering, two stages of motion filtering, and one stage of post-processing. The first accounts for variation in background, the next three for localization, and the final for error compensation. In particular, I contribute the utilization of optical flow as a second marker of hand location.

II. PROCESSING METHODOLOGY

Figure 1 illustrates the proposed processing pipeline. In the first step, an initial background model is computed and updated, and the video frames are color-balanced. The second step, drawing from [7], estimates the color of the subject’s skin to localize skin-colored regions within the image, while the third computes the difference between consecutive frames to estimate points of motion within the image. Step four uses the regions found above and calculates the motion found within them to estimate the position of the hands, and step five checks for errors in this estimate. These steps are detailed below:

A. Pre-processing

1) Background Model:

Drawing from [5], a background model is computed by averaging the first n frames of a video and then storing the result for later comparison. For each frame thereafter, the model is updated according to the formula:

This formula adapts quickly to newly introduced objects by giving the most recent frame greater weight, and by the same token, it quickly phases out moving objects that may be passing through the background. Figure 2 shows an example of a background model computed after the first n frames, with n=190.

2) Color Balancing:

Given the quality of cameras existing on mobile devices today, the proposed processing pipeline is benefitted by the addition of a color-balancing algorithm. Specifically, the gray-world algorithm (see Figure 3) from [3] was chosen here for its...
computational efficiency and general efficacy. Although this step should theoretically not be necessary for many applications because of the balancing performed by the camera, substantial experimental benefits were observed after its addition. Figure 4 shows a frame before and after color balancing.

\[ k_L \sum_{x,y} L[x,y] = k_M \sum_{x,y} M[x,y] = k_S \sum_{x,y} S[x,y] \]

*Figure 3: Defining equation of the gray-world algorithm*

**B. Color Filtering**

In conjunction with motion filtering, color filtering the video frames, as suggested by [6], [8], and [9], serves as one of the two primary mechanisms by which the proposed pipeline detects the hands. After color-balancing the image and removing the background using the current background model, Matlab’s Viola-Jones face detector from [4] is used, as in [8], to detect the face of the subject in the frame. For the present application, the face is assumed to be in the line of sight of the camera. After the face is detected, the jaw-cheek region is isolated in order to extract the hue, saturation, and value of this region. The jaw-cheek region, specifically, is chosen because it tends to have reduced variation in color when compared to the entire face, which includes the eyes, some hair, and some background. From these HSV values in the lower facial region, values for the median (for its greater robustness to noise) and standard deviation are calculated as descriptors for the skin color of the subject.

Using these descriptors, the frame is then scanned to isolate regions that match the skin color to within one standard deviation of each of the HSV median values. The image is then converted to grayscale, and using Otsu’s method for clustering, binarized. Following this, the video frame is eroded with a 3x3 square structuring element to reduce small amounts of noise, and then dilated with a 5x5 square structuring element to connect nearby regions, the result of which can be seen in Figure 5. The role of the dilation, in particular, is important because the bounding boxes of the remaining regions are calculated as initial hypotheses for the hand and arm locations. This image is stored as the color model.

**C. Frame Difference Filtering**

Concurrently with the color filtering, the current video frame is processed with respect to the background model. First, the absolute difference between the background model and the current frame is calculated to localize the subject region. The complement of the resulting image is then computed, which is then binarized, as in the previous step, by converting the image to grayscale and using Otsu’s method. This produces a rough edge map of small moving regions, and in particular, moving hands. Figure 6 illustrates this result. Subsequently, and as in the color filtering step, the image is eroded and dilated to reduce noise and connect nearby regions. This is shown in Figure 7.
size with respect to the size of the face) to potentially be a hand by comparing the region’s convex area with that of the face detected in the color filtering step. The selected bounding boxes are further filtered by comparing their color to that of the color found in the face, again, as in the color filtering step. Then any boxes within a certain distance of each other are merged into one (ie, the two center regions in Figure 5 would be merged). Finally, the remaining bounding boxes are saved as further initial hypotheses for hand and arm location.

Subsequently, the hypothesized bounding boxes resulting from the above filtering process, as well as the color filtering process are checked against each other. Specifically, only bounding boxes with non-empty intersection between the two binaries are kept. The key assumption herein is that there are no other regions in the image that favourably compare with the size of the face, that are approximately skin colored, and that are moving substantially in the video. These are considered to be the strongest hypotheses for the hand and arm regions of the subject.

D. Localized Motion Filtering

After the above filtering process has finished, few hypotheses should remain for the hands and arms. In experiments, there were, in fact, two or fewer. Should there be more, a simple heuristic is used to select the two bounding boxes that are closest to the previously estimated location of the hands and that are of approximately appropriate size relative to the size of the subject’s face.

Once there are two or fewer hypotheses for the hands, optical flow within each hypothesized region is calculated. In particular, the Lukas-Kanade method is used, whose formula is given in Figure 8, to extract the pixel with largest magnitude optical flow.

The motivation behind extracting these largest magnitude pixels is that, generally, within the hypothesized bounding boxes, if there is motion, it will be concentrated around the edges of the hand. Given this, averaging the pixel positions to produce the centroid should give a good estimate of the hand position. The results of this are shown in Figure 9.

\[
\begin{bmatrix}
\sum_{x,y \in W} f_x^2 & \sum_{x,y \in W} f_x f_y \\
\sum_{x,y \in W} f_x f_y & \sum_{x,y \in W} f_y^2
\end{bmatrix} \begin{bmatrix}
\Delta x \\
\Delta y
\end{bmatrix} = \begin{bmatrix}
\sum_{x,y \in W} f_x f_t \\
\sum_{x,y \in W} f_y f_t
\end{bmatrix}
\]

Figure 8: Solution to Lukas-Kanade algorithm.

E. Post-processing

Having identified the approximate location of the hands, some post-processing must be done in order to avoid several frequent errors.

First, as noted previously, the above process takes the intersection of the binary images resulting from color filtering and difference filtering. As a consequence of this and the fact that the motion filter will not yield an identifiable region when the hands are motionless (see Figure 10, which shows the exact result), the process as thus far described will fail to produce a location for the hands once they stop moving. As a simple approximation to the location in this case, the previously identified location is simply carried over to the new frame.

Second, it will occasionally be the case that the hands come together to touch. In this case, only one bounding box is found and the above method fails to identify the hands. A special case is made for this occurrence, though, so that the coordinate found within that one bounding box is made to represent both hands. Although there will be greater error in this case, the difficulty involved in distinguishing the hands is beyond the scope of this report and outside the interests of its current application.

Third, although the assumptions required by the color and frame difference filtering, namely that there be no other moving, skin-colored regions in the video, generally hold, they occasionally do not and may produce a dramatically incorrect result. In order to approximately, though certainly not perfectly, correct for this, when the found coordinates of the hands differ substantially from the coordinates found in the previous frame, the result is assumed to be invalid and the location from the previous frame is simply carried over to the current frame. This is marked in the sample video by an
absence of ‘X’ optical flow markers. Although this excludes the possibility of the hands moving very rapidly, it represents a necessary trade-off in order to reduce error. Further, when the bounding boxes found after the color and frame difference filtering overlap considerably, the bounding boxes are assumed to describe the same region and are merged into one region. Finally, once the coordinates of the hand are calculated for the current frame, they and the new background model are stored in order to propagate them forward to the next frame.

III. OBSERVED AND HYPOTHESIZED ERRORS

The most commonly observed error concerns the distribution of largest magnitude pixels within the hypothesized hand bounding boxes. In particular, the proposed methods treat the hand and arm (assuming the arm’s skin is exposed) as one unit. This is a consequence of the fact that when the hand is moving, the arm often is as well, and so the color and frame difference filters detect both objects. In many situations, this is not a problem, as the hand is moving as fast as, if not faster than, the arm and so the largest magnitude pixels remain concentrated around the hand. Sometimes, however, this is not the case and the computed centroid will be located closer to the midpoint between the hand and elbow, as in Figure 11. As long as this new position is not further from the previous one than the distance heuristic described in the section on post-processing allows, however, a more accurate position should be quickly, though not immediately, recovered.

Figure 11: Shortcoming of the proposed detector in the case where the motion in the hands does not exhibit the greatest (magnitude) motion

A second problem arises when a combination of two conditions is present: namely, that the skin-colored regions of the subject (namely the arms and hands) are overlapping, but the hands are separated, as in Figure 12. Because there is very little that remains, at this point, to differentiate the two hands or arms without depth information, it is difficult to separate the two and so the processes proposed above, depending on the extent of the overlap of the bounding boxes of the hands and arms will either be merged into one box and the location of the hands is estimated normally (though incorrectly, note), or they will not be and the calculated optical flow may be affected by the motion of the external arm or hand within the bounding box of the other.

Figure 12: Shortcoming of the proposed detector when the bounding boxes intersect substantially, yet the hands remain distant.

Perhaps the most substantial problem, however, arises when the hands cross each other with respect to their x-coordinate, and pass closely with respect to their y-coordinate. In this case, the identifier for each hand can be swapped as a result of the fact that the heuristic for such an event is assigning the coordinates to whichever hand minimizes the total distance between the current coordinates and those of the previous frame. That is, suppose hand 1 was at \((x_1, y_1)\) in the previous frame, and hand 2 was at \((x_2, y_2)\) in the previous frame and that the new coordinates for are \((x_1', y_1')\) and \((x_2', y_2')\). Then \((x_1', y_1')\) and \((x_2', y_2')\) will be assigned so as to minimize the distance between the non-primed and primed coordinates. Although this problem can become serious when the temporal component of the hand location is vital, for applications less dependent on the temporal coordinate or symmetric with respect to both hands, this may not be a problem at all. Moreover, it is a fairly uncommon problem, so much so that it could not be reproduced.

IV. FUTURE WORK

In continuing this processing pipeline, several improvements can be made. Most important among these is developing a more reliable method of extracting hand position once the bounding box for the hand and arm region has been determined. If the subject is wearing a long sleeve or jacket, theoretically, none of the aforementioned problems, aside from self-occlusion, should be observed. In fuller generality, though, this problem is substantial. Another further step would consist in extracting high-detail information from the hands. As it is, only the rough position of the hands is calculated. From this, an approximate bounding box for the hand can be determined so as to extract information about finger positioning. However, the proposed processing pipeline fails to provide coordinates for hand and finger joints, account for finger position, or provide information about hand and finger orientation. This is an extremely challenging problem as a result of the number of degrees of freedom exhibited by the hand and, in fact, remains to be solved in the study of gesture recognition.

V. CONCLUSION
In consideration of the constraints placed on this problem – namely of minimizing computational cost and utilizing only past frame and RGB information, the proposed processing pipeline performs fairly well, accounting for many forms of gesture. Although it assumes that there are no other moving, skin-colored regions in the image that remain after background subtraction, and that within the regions found by color and frame difference filtering those pixels with greatest magnitude optical flow are distributed around the hand, it is reasonable to assume that these conditions will be satisfied for, at least, the application of large gesture tracking. Further, the addition proposed by this pipeline, namely the use of optical flow in helping to determine hand location, performs reasonably well in most instances, as can be seen in the provided sample video. Nonetheless, much work remains in more cleanly handling instances self-occlusion, in generalizing the environmental conditions under which the pipeline will produce valid results, and in extracting detailed information from the hand once its location has been determined.

ACKNOWLEDGEMENT

I would like to thank Professor Bernd Girod of Stanford University’s Department of Electrical Engineering for his expertise in the methods used, as well as, more general, in the subject of digital image processing. I would also like to thank the EE 368: Digital Image Processing’s TAs and advisors David Chen, Matthew Yu, and Andre Araujo, also at Stanford University, for their support and mentorship.

REFERENCES


