DETECTING AND CLASSIFYING FRONTAL, BACK AND PROFILE VIEWS OF HUMANS

Narayanan Chatapuram Krishnan, Baoxin Li and Sethuraman Panchanathan
Center for Cognitive and Ubiquitous Computing, Arizona State University, Tempe, USA

Keywords: Human Detection, graph cut, shape context, SVM.

Abstract: Detecting and estimating the presence and pose of a person in an image is a challenging problem. Literature has dealt with this as two separate problems. In this paper, we propose a system that introduces novel steps to segment the foreground object from the back ground and classifies the pose of the detected human as frontal, profile or back view. We use this as a front end to an intelligent environment we are developing to assist individuals who are blind in office spaces. The traditional background subtraction often results in silhouettes that are discontinuous, containing holes. We have incorporated the graph cut algorithm on top of background subtraction result and have observed a significant improvement in the performance of segmentation yielding continuous silhouettes without any holes. We then extract shape context features from the silhouette for training a classifier to distinguish between profile and nonprofile(frontal or back) views. Our system has shown promising results by achieving an accuracy of 87.5% for classifying profile and non profile views using an SVM on the real data sets that we have collected for our experiments.
.

1 INTRODUCTION

Detecting the presence of a human and estimating his/her pose in a video sequence is a challenging problem and has been an active area of research,witnessing a surge of interest in the recent years with widening spectrum of applications. Our motivation behind this problem is designing a system that can detect the presence of a human and also estimate the pose, that can be used as an assistive device for individuals who are blind in their office spaces. We hope that such a system would be useful in informing an individual who is blind, if there are any people standing by the entrance of their office space. The requirement of such a system makes it different from other systems for surveillance.Apart from being accurate in detecting a person at the entrance, information about the pose of the person(frontal,back or profile) is useful for the individual who is blind to judge whether the person is waiting for them to respond.

In our proposed system, background subtraction is performed on the frames obained from a stationary video camera. On detection of significant change, the system localizes the change and performs a robust segmentation to obtain the silhouette of the region that has changed. This silhouette is then used to determine whether the moving object is a human and to estimate pose of the person. We envision conveying this information to the individual who is blind, through an audio device. It is difficult for individuals who are visually impaired to come to know if there are people standing by their door side. We hope that such a system will enhance the social interaction ability of the individual by giving them cues about people who are passing by and who are standing by their door.

We have divided this paper in the following manner. Section 2 gives a summary of the related work on human detection. The details about background subtraction and foreground segmentation are discussed in section 3 and the classification step in section 4. Section 5 presents the results obtained at the various stages and analyses them. Conclusions and future work are presented in sections 6.

Chatapuram Krishnan N., Li B. and Panchanathan S. (2007).
DETECTING AND CLASSIFYING FRONTAL, BACK AND PROFILE VIEWS OF HUMANS.
In *Proceedings of the Second International Conference on Computer Vision Theory and Applications - IU/MTSV*, pages 137-142

2 RELATED WORK

Background subtraction techniques for human detection rely on the static background information or on the motion information, to first detect the regions that have changed or moved. Beleznai etal (Beleznai et al., 2004) perform a Mean shift clustering of the subtracted image to identify regions of significant change. The clustered regions are then checked for the presence of humans by fitting a simple human model in terms of three rectangles. Eng and etal (Eng et al., 2004) propose a similar technique where the local foregrond objects are detected using clustering, and an elliptical model is used to represent the humans. A Bayesian framework is then employed to estimate the probability of the presence of a human after fitting the ellipses to a foreground object region. Elzein and etal (Elzein et al., 2003) propose a vision based technique for detecting humans in videos. A localized optic flow computation is performed to compute the locations that have undergone significant amount of motion. Haar wavelet features at different scales are extracted from these localised regions, and matched against that of templates using a linear classifier. Lee and etal (Lee et al., 2004) use differential motion analysis to subtract the current input image from a reference image and thus extract the contour of the moving object. A curve evolution technique is then performed to remove the redundant points and noise on the contour. The curve thus extracted is matched against existing templates by calculating the Euclidean distance of the turn angles at the points describing the curve.

Dalal and etal (Dalal. Navneet and Schmid, 2006) propose a technique for detecting the humans based on oriented histrograms of flow and appearance. Optic flow is computed between successive frames and the direction of flow is quantized. The histogram constructed based on these directions of flow, coupled with the histograms of oriented gradients are used to train a linear SVM to detect the presence of humans. Bertozzi and etal (Bertozzi et al., 2005) describe a system for pedestrian detection using stereo infrared images. Warm areas from the images are detected and segmented. An edge detection operation is performed on the resulting regions, followed by a morphological expansion operation. Different head models are then used to validate the presence of humans in the resulting images. Researchers have proposed similar algorithms (Zhou and Hoang, 2005; Han and Bhanu, 2005) to detect humans using either motion information or static background information. But all of these algorithms, have stopped at detecting whether the moving object is a human or not. We go a step further and also provide information about the high level pose of the person by indicating whether it is a frontal, back or profile view of the person. As mentioned before, estimating the pose of the person will help an individual to decide if the person at the door is waiting for them to respond or not.

We employ a background subtraction technique, enhanced by graph cut algorithm to segment the foreground regions. Silhouettes thus obtained are used to extract features like shape context and fourier descriptors. These are then used to train a classifier to distinguish between the profile and non profile views. Though many researchers use synthetically generated human silhouettes for testing the pose estimation algorithms, we have used real segmented silhouettes for evaluating our pose estimation algorithm.

3 SEGMENTATION

This section deals with the 3 stage process of extracting the silhouette of the foreground objects. A running average model is based on technique proposed by Wren etal in (Wren et al., 1997)piccardi:04, where the background is independently modeled at each pixel location is used to model the background. A Gaussian probability density function (pdf) that fits the pixel's last n values is computed. A running average is computed to update the pdf. Often, even with these models, the shadow regions get misclassified as foreground. Assuming that the illumination component of the pixel locations in the shadow region undergo uniform change, we use the derivatives at these locations to cancel out this unform change. The derivatives of the pixel locations in the shadow region for both the background model and the current frame should be very similar. Thus the difference in the derivatives can help in eliminating to a certain extent the shadow regions.

Robust background subtraction can still result in broken contours and blobs of the foreground object. It is difficult to detect whether the foreground object is a human using these blobs. Ideally we would like to have a continuous contour that can be further processed. To obtain this continuous contour we have used the extended version of graph cut algorithm (Boykov and Jolly, 2001) proposed by Rother etal (Rother et al., 2004) for color image segmentation. Though this approach guarantees an optimal segmentation solution, given the constraints, the drawback is that the seed or the trimap(the initial

foreground, background and unknown regions) has to be manually initialized. We have improved upon this framework by automatically constructing the trimap from the background subtracted mask obtained in the previous step. We explain briefly the graph cut framework and refer the reader to (Boykov and Jolly, 2001; Rother et al., 2004) for the theoretical and implementational details.

The trimap consists of three regions namely, background, foreground and unknown. Gaussian mixture models (with K components) are computed for the background and foreground pixel classes using the minimum variance color quantization approach proposed by Orchard and Boumann (Orchard and Bouman, 1991). Each pixel in the foreground set is assigned to the foreground GMM component that has the highest likelihood of producing that color. Similarly the background pixels are also assigned to the most likely background gaussian component. The Gaussian mixtures are then recomputed from the newly created pixel sets. A graph is constructed as described in (Boykov and Jolly, 2001). Every pixel in the image is associated with a node in the graph along with two other special nodes- the foreground and the background node. These nodes are joined by two types of links - N-links that connect every pixel with its 8 neighbors and the T-links which connect every pixel to the foreground node and the background node. The N-link describes the penalty for placing the segmentation boundary between neighboring pixels, which is set to be high in regions of low gradient and low in regions of high gradient. The T-links describe the probability of each pixel belonging to the foreground or to the background. The weight for the N-link between pixel i and j is given by

$$W_n(i,j) = \frac{\gamma}{dist(i,j)} e^{-\beta \|I_i - I_j\|^2}. \quad (1)$$

where I_i is the color of pixel i and $dist(i,j)$ is the Euclidean distance between pixel locations i and j. (Rother et al., 2004; Boykov and Jolly, 2001) suggest setting $\gamma = 50$ and β as given in equation 2 .

$$\beta = \frac{1}{2\left\langle \|I_i - I_j\|^2 \right\rangle} \quad (2)$$

The T-link weights are computed as described in table 1,where $L(i) = 8\gamma + 1$ and $W(i)$ is the log likelihood of a pixel belonging to either background or foreground given by

$$W(i) = -log \sum_{k=1}^{K} \pi_k \frac{1}{\sqrt{det\Sigma_k}} e^{\left(-\frac{1}{2}[I_i - \mu_k]^T \Sigma_k^{-1} [I_i - \mu_k]\right)} \quad (3)$$

Table 1: Weights for the T-Links.

Pixel Type	Background	Foreground
$i \in$ foreground	0	L(i)
$i \in$ background	L(i)	0
$i \in$ unkown	$W_{fore}(i)$	$W_{back}(i)$

Thus once the weights for all the links are computed, the problem reduces to finding the cut, that maximises the flow from the foreground node to the background node. This is performed by the fast mincut algorithm implemented by (Boykov and Jolly, 2001).The final result of the segmentation process is an optimal solution to this minimizing energy problem.

One of the main disadvantages of this technique as pointed out by Kumar etal (Kumar et al., 2005) is that this framework does not have a mechanism to segment out natural shapes. Keeping this in mind, we have formulated a simple way of generating the trimap from the background subtracted mask. We preserve the shape of the segmented mask,the foreground region of the trimap, as much as possible along with making the holes within the mask and a thin boundary around the mask as unknown regions of the trimap.

4 SHAPE CLASSIFICATION

Our objective behind using shape classification is two fold - categorize the silhouette as a human silhouette and to estimate whether it is the profile non profile view of the person. As mentioned before our objective is not to estimate the accurate pose of a person, as dealt by others. We have used the shape context feature proposed initially by Belongie and etal (Belongie et al., 2002) for shape classification and later modified by Agarwal and Triggs (Agarwal and Triggs, 2004) for pose estimation. The silhouette is first processed to select n contour points at equally spaced intervals. The shape context at a point describes the spatial locations of the other $n-1$ sampled points with respect to the point under consideration in a histogram. The shape is encoded as a distribution in the 60-D shape context space(12 angular bins and 5 radial bins). Belongie and etal (Belongie et al., 2002) match the shape context vectors extracted from two silhouettes, using a bipartite weighted graph matching algorithm, while Agarwal and Triggs (Agarwal and Triggs, 2004) compute another layer of histogram before applying the relevance vector regression for pose estimation. We have taken the step of generating a second layer of histograms.The distribution of all the points on a silhouette is reduced

to 100-D histograms by vector-quantizing the shape context space. The 100 center codebook is learnt using a K-means clustering algorithm over all the 60-D shape context vectors in the training set. The 100-D final histogram is then constructed by allowing every shape context vector of a silhouette to vote softly using gaussian weights into the bins corresponding to the top 5 nearest centers to them and subsequently accumulating it over all the points in a silhouette.

Instead of using relevance vector regression as in (Agarwal and Triggs, 2006) or having exemplars to denote the different shapes (Poppe and Poel, 2006), we experimented with training a support vector machine (SVM), for distinguishing between different views of a person. The SVM was trained with the 100-D histograms extracted out of the silhouettes from the training set. It is intutive that the silhouette of the frontal and the back view of a person are very similar. Thus trying to distinguish between these two views using silhouettes may be a futile effort. However, the silhouettes of the profile view are significantly different from the frontal or back. Thus at this step, we have tried to differentiate between the profile view from the non-profle view. The training set consists of the feature vector extracted from the frontal as well back views in one class and that of the profile view in another class. In the next step as described in the following subsection, we classify the non-profile view as frontal or back based on the image data.

Frontal Vs Back View distinction: As it is difficult to differentiate between the frontal and back view of a person using the silhouette information, we revert to the color information present in the image. We have used a skin color detection algorithm, to compute the number of pixels in the upper region of the silhouette, corresponding to the head region, having the skin color. If this number is greater than a threshold, we classify the view as frontal else we label it as back.

5 RESULTS AND DISCUSSION

Though there are existing synthetically generated clean human silhouette databases (Agarwal and Triggs, 2004),it is important to work with real data to understand the problems and limitations, encountered when a vision based system is deployed in a real environment. We captured people entering, exiting and passing by a cubicle in our lab using a sony handycam. 10 second videos with a frame rate of

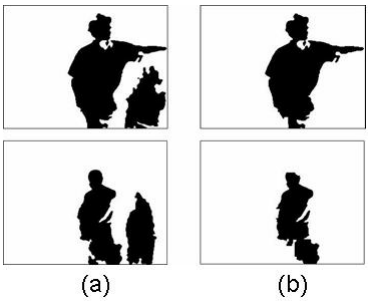

Figure 1: The results of the shadow removal step.

30 seconds of 15 individuals under varying lighting conditions were captured. Individual frames from the video sequence were then extracted and manually labeled as either belonging to frontal, back or the profile view of the person. A total of 604 frontal, 297 back and 277 profile view images, on an average of 40 frontal, 20 back and 20 profile views per person were thus collected and labelled. The first row of Figure 5 shows some of the sample frames.

Figure 5 presents the results obtained after the shadow removal step. Column (a) contains the images obtained after background subtraction and column (b) is the result obtained after combining (a) with the difference of the response to the LoG operator. Some regions, in the image in the second row that belonged to the foreground object also got erased after the shadow removal step. However, the graph cut algorithm in the next step, rectifies the segmentation by including these regions in the foreground object as shown in the image in the last row of column (e) in figure 5. This is possible because, regions around foreground silhouette in the image shown in 5(b) are considered as unkown regions when constructing the trimap for the graph cut algorithm.

Figure 5 illustrates the process of segmentation. Each column contains the results obtained at the different steps in the segmentation process of the image shown in the first row of the column. One important observation about the final segmentation result(last row in figure 5), is that the end results have no holes, and the silhouette is continuous, when compared with the back ground subtracted image shown in the second row. The third row depicts the trimap that is created out of the image in the second row. As it can be seen, the overall shape of the silhouette is more or less preserved in the trimap. It is evident from the results that adding the graph cut step indeed improves the segmentation result, thereby

Table 2: Comparison of accuracies for different values of the parameters - number of code vectors and number of sample points.

		Number of Code Vectors		
		50	100	150
Number of Sample Points	50	86.64%	85.45%	86.11%
	100	86.54%	87.5%	86.93%
	150	86.25%	87.5%	87.14%

Table 3: Comparison of the performance of Shape context and Fourier descriptors as features for classifying the silhouette as profile or non profile.

Feature Vector	Linear SVM	Gaussian SVM
Shape Context	87.5%	83.1%
Fourier Descriptors	76.53%	73.6%

improving the classification accuracy.

After the silhouettes are computed as shown in the last row of figure 5, shape features are extracted from it. We have experimented with two features described in the literature - Shape Context and Fourier Descriptors. The shape context features were implemented by considering 100 equally spaced points on the silhouette boundary. The larger the number of sampled points, the more accurate is the silhouette description. However, the noise in the contour as a result of segmentation process also gets encoded. Similarly, the lesser the number of sample points, lesser is the accuracy of the encoded silhouette. We experimented with 50, 100 and 150 sample points for describing the silhouettes. The classification accuracies obtained are shown in table 2. As mentioned before a second layer of histogram is computed to further quantize the shape context vector. The auhtors (Agarwal and Triggs, 2004) cluster the training shape context vectors in to 100 code vectors. We experimented with different number of code vectors. In both the cases it can be noted that there is no significant change in the performance.

Translational, rotational and scale invariant fourier descriptors were computed as mentioned by (Poppe and Poel, 2006). However instead of using exemplars to compare the test silhouette, we trained an SVM classifier. The best performance with fourier descriptors for classifying the silhouettes as profile or non-profile view was around 76.53%, while an accuracy of 87.5% was achieved using shape context features. We have used the leave one out strategy, where for every fold, images from one video was considered as the test sequence and the remaining were the training sequence. This clearly shows the superiority of shape context features over fourier descriptors in the current context. There was no significant change in the performance of the features when a gaussian SVM was used instead of a linear classifier. These results are summarized in table 3.

We used the standard skin tone detection approach to distinguish between the frontal and the back views. Depending on the size of the silhouette, regions from the top part of the silhouette were extracted for detecting the presence of skin tone. A threshold was determined for classification. We were able to get an accuracy of 71%. One of the reasons for getting a low accuracy was that, regions other than the head region also got included while computing the percentage of skin color. We are working on techniques that would consider only the elliptical head region to determine the percentage of skin color.

6 CONCLUSION

We have proposed a system for detecting the presence of a human in an indoor environment and classifing the pose at a high level as frontal, back or profile view. We have improved on the traditional background subtraction method, by adding a step that performs graph cut. The results obtained after this step show significant improvement over the background subtracted images. The silhouette thus extracted was used for classiyfing the profile and non profile views of the human. We experimented with two features for this purpose - shape context and fourier descriptors. We observed that shape context features perform significantly better(with an accuracy of 87.5%) than the fourier descriptors(with an accuracy of 76.53%) in classifying a silhouette as profile or non profile. We further have used the percentage of skin color to classify the non profile view as either frontal or back, achieving around 71% accuracy. We intend to use this as a front end to an intelligent environment we are developing to assist individuals who are visually impaired in their office spaces.