# A MODEL-BASED APPROACH FOR HUMAN HEAD-AND-SHOULDER SEGMENTATION

*Xiaowei Deng*[*], *Yuxiang Shen*[†], *Xiaolin Wu, IEEE Fellow*[*] *and Liang Zhao*[†]

Hulu LLC[†],
McMaster University[*]

## ABSTRACT

Object boundary extraction has long been a fundamental research topic, as well as an essential component in many visual computing and communication algorithms, such as computer vision, robotics, pattern recognition and video compression. Under this topic, human head-and-shoulder segmentation is of particular meaning, given the ubiquity of head-and-shoulder type of videos in social media, teleconferencing, and entertainment. Although human visual system can easily detect and recognize the head and upper body of a person, this seemingly simple task still poses a challenge to computers. In this paper, an effective and efficient segmentation method is proposed. This method consists of a novel human body descriptor in polar coordinates and a Markov chain based boundary model, which work together to generate precise boundary results. Moreover, dynamic programming is employed in this work, so as to accelerate the segmentation process. Comparisons with other algorithms are made in the experimental part, which clearly exhibits the advantage of our proposed method over some of its precedents.

***Index Terms***— Human body model, head-and-shoulder segmentation, polar coordinate system, dynamic programming, Markov chain

## 1. INTRODUCTION

Identifying, tracking and extracting objects out of a complex background in images and videos is a common, important and challenging problem that arises in many technical fields, including computer vision, robotics, pattern recognition, low-bitrate video compression, visual communication and consumer electronics. In this paper, we focus on a particular segmentation task: extracting and representing the head-and-shoulder boundary of humans, which we call the head-and-shoulder object (HSO), through video frames. Our research is motivated by the ubiquity of head-and-shoulder type of videos in social media, teleconferencing, and entertainment. Although the problem can be simply stated, its solution is far from being straightforward as it might appear. In fact, until now no published video segmentation algorithms can solve the HSO segmentation problem with sufficient precision and robustness to be free of visible artifacts; it is very difficult for the video segmentation algorithm to withstand human or/and camera motions against complex background, varying illuminations, and noises.

Many image/video segmentation techniques are available in the literature [2-11]. Early techniques tend to use region splitting or merging [3,6,7], which correspond to divisive and agglomerative algorithms in the clustering literature. More recent algorithms [8-11] often optimize some global criterion, such as intra-region consistency and inter-region boundary lengths or dissimilarity. Recently, there are some works specially designed for the HSO segmentation problem [9,11]:

In this paper, we propose an automatic head-and-shouler segmentation method, which can extract head-and-shouler regions effectively and accurately from input images/video sequences.

The proposed algorithm, as schematically represented by the flowchart in Figure 1, comprises the following steps: 1. Face localization by finding the landmark points of eyes and mouth and representing the face in a polar coordinate system. 2. Anchored on the face landmark points, locating the other parts of the compounded head-and-shoulder object (HSO): head, neck and shoulders. 3. A graph-based segmentation process is proposed to extract the boundary. The segmentation is formulated as a global optimization problem and solved by dynamic programming. 4. Combining all segmented parts together to obtain the final head-and-shoulder segmentation result.

The rest of paper is organized as follows: in the next section, the polar coordinate system is introduced to model the human body; Section 3 formulates HSO boundary segmentation into an optimization problem using Markov Chain; A dynamic programming style solution is presented in Section 4 to search for the optimal HSO boundary. Section 5 provides some experimental results and Section 6 concludes this paper.

## 2. GEOMETRIC REPRESENTATION OF HSO

In order to identify and segment the HSO, we first locate the person's face in the image [1]. The detected face is used as a base to build the HSO by locating and integrating the other peripheral parts: hair, neck and shoulders.

In our HSO segmentation algorithm, a polar coordinate system is introduced to represent the face. Compared to

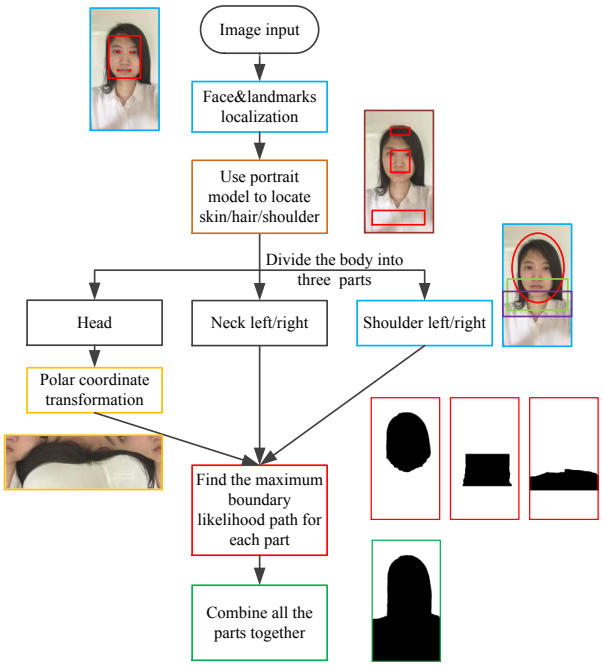


**Fig. 1**: Flowchart of general head-and-shoulder segmentation

Cartesian coordinates, the polar face representation can better characterize the in-plane face rotations; it is easier to describe the contour of a face using polar coordinates as human face is roughly round thus its contour is approximately a line in polar coordinate systems. Once we can anchor the centre of the polar coordinates system using facial features, it is easy to mark those objects that could have irregular shapes, such as ears, so that the algorithm can compensate accordingly which is discussed in Section 3 and Section 4.

Now, we introduce the polar coordinate system with respect to the head position as follows; 1. Original Point: The eyes sit at the vertical center of the head or just above, about halfway between the top of the skull (not the hairline) and the bottom of the chin. The middle point of the two eyes can be the origin of the polar coordinate system, playing the role of anchoring the shape descriptor; 2. Base Axis: A line fitting two or more predetermined eye points is chosen to be the base axis of the polar coordinate system, playing the role of calibrating the orientation of the object;

Intuitively we apply an ellipse model to fit the head: After calibrating the distance between two eyes and the distance between eyes and mouth, we can approximately find out the major radius and minor radius of the head ellipse model (as shown in Figure 2, where ellipse with dotted line is the head model).

The actual boundary of head, which is what we want, is close to the head ellipse model we just presented. So instead of searching the entire image, we can just simply search the boundary in the narrow ellipse band, as shown in Figure 2.

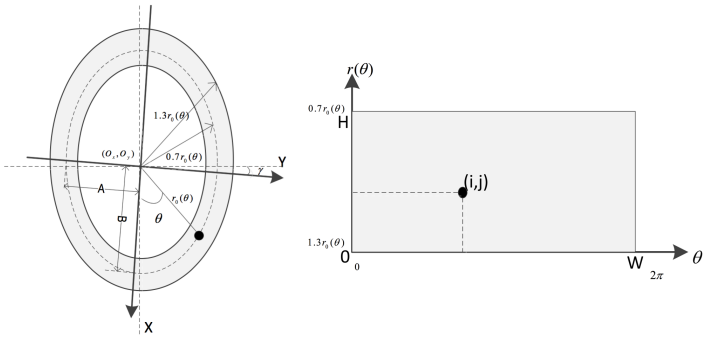


**Fig. 2**: Mapping ellipse band to square

Then we use the coordinate transformation formula to map the gray ellipse band area into $W \times H$ rectangular to normalize the size of the model. This ellipse model under polar coordinate system is largely invariant to translation, in-plane rotation and scaling of the head image that are caused by movements of the head and/or camera because the skull is a rigid body and the relative positioning of the corner points of the two eyes on a given face is fixed.

Human face and body are most familiar things in our life and we should make full use of its structure as an prior knowledge. After face localization and landmarks detection, we divide the HSO into three parts: head, neck and shoulder. Our boundary search algorithm will be applied on each part separately.

The structural model of head-and-shoulder is based on these principles of proportion: 1. The distance from chin to shoulder line is roughly 1/3 of height of head. 2. The width of shoulders is roughly twice of height of head.

The purpose for this partition is to reduce the complexity of boundary search by making all the boundaries for each part become a one-dimensional left-to-right signal, which is much easier for computers to process, classify and recognize. For example, for the head part, the head boundary can be transformed into a polar coordinate system (as described in Section 2 ); for the neck part, the boundary can be rotated by 90°/-90° . Then within each part we utilize the boundary extraction algorithm (to be detailed in Section 3) to get the final result.

## 3. GRAPH-BASED HSO SEGMENTATION

Before we introduce the graph-based algorithm for HSO segmentation, we construct a Gaussian Mixture Model for the HSO. This model will be used to estimate the probability of boundary later.

Generally HSO consists of 3 textures: skin(face,neck), hair and clothes. It is not hard to collect some sample data from each part. Statistically we can find that the data of each part can be viewed as an Gaussian Mixure Model. Intuitively

we assume that there are 2-3 Gaussians for each texture. For example, the clothes may have 3 colors, the person may wear glasses on face, etc.

Fitting the data with Gaussian Mixture Model in each class can be solved via the standard EM algorithm. This provides us the likelihood of RGB vector $\boldsymbol{x}$ given each class: $\boldsymbol{\mathcal{P}}(\boldsymbol{x}|class), class \in \{\text{skin, hair, clothes}\}$.

To find out the posterior probability $\boldsymbol{\mathcal{P}}(class|\boldsymbol{x})$, we use the Bayes rule:

$$\boldsymbol{\mathcal{P}}(class|\boldsymbol{x}) \propto \boldsymbol{\mathcal{P}}(\boldsymbol{x}|class) \cdot \boldsymbol{\mathcal{P}}(class) \tag{1}$$

where $\boldsymbol{\mathcal{P}}(class)$ is the prior probability of each class. We can simply approximate them with equal probability 1/3.

Finally, the HSO probability is defined by the maximum likelihood in three texture classes:

$$\boldsymbol{\mathcal{P}}(\text{HSO}|\boldsymbol{x}) = \max\{\boldsymbol{\mathcal{P}}(skin|\boldsymbol{x}), \boldsymbol{\mathcal{P}}(hair|\boldsymbol{x}), \boldsymbol{\mathcal{P}}(clothes|\boldsymbol{x})\} \tag{2}$$

After we get the approximate position of HSO and divided it into several parts based on prior knowledge of HSO structure. Within each part, we apply a graph-based algorithm to extract the boundary.

The basic idea of our proposed graph-based algorithm is to build a weighted undirected graph $\mathcal{G} = (V, E)$, with the set of vertices $V = \{v_{i,j}, 1 \le i \le W, 1 \le j \le H\}$ (image size is $W \times H$) corresponding to the pixels in the image. Edges $E$ in the graph occur between any two nodes $v_{i,j}$ and $v_{m,n}$ close to each other. The edge weight $w(v_{i,j}, v_{m,n}) \ge 0$ reflects the likelihood of the edge to be HSO boundary.

Let $\boldsymbol{\mathcal{B}} = \{b_1, b_2, ..., b_W\}$ denote the edges set of HSO boundary, where $b_i$ is the edge between nodes $v_{i,m}, v_{i+1,n}, m, n \in \{1, ..., H\}$. Let $\boldsymbol{X}$ be the input image. We fomulate the computation of HSO boundary $\boldsymbol{\mathcal{B}}$ as the following MAP estimated problem.

$$\begin{aligned} \boldsymbol{\mathcal{B}}_{\text{MAP}}(\boldsymbol{X}) &= \arg\max_{\boldsymbol{\mathcal{B}}} \mathcal{P}(\boldsymbol{\mathcal{B}}|\boldsymbol{X}) \\ &\propto \arg\max_{\boldsymbol{\mathcal{B}}} \mathcal{P}(\boldsymbol{X}|\boldsymbol{\mathcal{B}})\mathcal{P}(\boldsymbol{\mathcal{B}}) \end{aligned} \tag{3}$$

where the term Boundary Model $\mathcal{P}(\boldsymbol{\mathcal{B}})$ reflects the probability of a series of edges $b_1, b_2, ..., b_W$ of the input image part that align from left to right. The term $\mathcal{P}(\boldsymbol{X}|\boldsymbol{\mathcal{B}})$ is the probability of input image under the specific boundary pattern.

Here we suppose edge sequences $\boldsymbol{\mathcal{B}}$ follow a boundary markov process. Then we can get,

$$\begin{aligned} \mathcal{P}(\boldsymbol{\mathcal{B}}) &= \mathcal{P}(b_1, b_2, ..., b_W) \\ &= \mathcal{P}(b_1)\mathcal{P}(b_2|b_1)\mathcal{P}(b_3|b_2)...\mathcal{P}(b_W|b_{W-1}) \end{aligned} \tag{4}$$

The initial probability $\mathcal{P}(b_1)$ could be assigned 1. One fact is that the boundary is continuous from left-to-right, so the two adjacent edges should be connected. Another fact is that the boundary is relative smooth, so the slope should not change too much either.

The probability of $\mathcal{P}(b_{i+1}|b_i)$ can be estimated as follows:

$$\mathcal{P}(b_{i+1}|b_i) = \begin{cases} 0 & \text{if } b_{i+1}\text{ , } b_i \text{ not connected} \\ e^{-\frac{|\theta(b_i) - \theta(b_{i+1})|}{\sigma}} & \text{otherwise} \end{cases} \tag{5}$$

where $|\theta(b_i) - \theta(b_{i+1})|$ is an estimation for slope difference between $b_i$ and $b_{i+1}$.

Since the edge $b_i$ is only affected by surrounded pixels, we only focus on the boundary image patches $\boldsymbol{X}_{b_1}, \boldsymbol{X}_{b_2}, ..., \boldsymbol{X}_{b_W}$, where $\boldsymbol{X}_{b_i}$ is the image patch around edge $b_i$. Suppose that only pixel patches near the HSO boundary $\boldsymbol{\mathcal{B}}$ depend on $\boldsymbol{\mathcal{B}}$, then we have

$$\begin{aligned} \mathcal{P}(\boldsymbol{X}|\boldsymbol{\mathcal{B}}) &= \mathcal{P}(\boldsymbol{X}_{b_1}, \boldsymbol{X}_{b_2}, ..., \boldsymbol{X}_{b_W}|b_1, b_2, ..., b_W) \\ &= \mathcal{P}(\boldsymbol{X}_{b_1}|b_1)\mathcal{P}(\boldsymbol{X}_{b_2}|b_2)...\mathcal{P}(\boldsymbol{X}_{b_W}|b_W) \end{aligned} \tag{6}$$

Each factor $\mathcal{P}(\boldsymbol{X}_{b_i}|b_i)$ is a local image patch likelihood given the edge. We use 2 terms to estimate each factor:

$$\mathcal{P}(\boldsymbol{X}_{b_i}|b_i) = \mathcal{P}_{\boldsymbol{\mathcal{E}}}(\boldsymbol{X}_{b_i}|b_i) \cdot \mathcal{P}_{\boldsymbol{\mathcal{B}}}(\boldsymbol{X}_{b_i}|b_i) \tag{7}$$

where $\mathcal{P}_{\boldsymbol{\mathcal{E}}}(\boldsymbol{X}_{b_i}|b_i)$ reflects the probability for $b_i$ to be an edge. $\mathcal{P}_{\boldsymbol{\mathcal{B}}}(\boldsymbol{X}_{b_i}|b_i)$ is the probability for $b_i$ to be a boundary of HSO.

$\mathcal{P}_{\boldsymbol{\mathcal{B}}}(\boldsymbol{X}_{b_i}|b_i)$ can be defined using HSO probability. Since the boundary region should exist in patches where the inner area is foreground and the exterior area is background, the higher the contrast between inner and exterior area, the more likely $b_i$ is to be the boundary.

$$\begin{aligned} &\mathcal{P}_{\boldsymbol{\mathcal{B}}}(\boldsymbol{X}_{b_i}|b_i) \\ &= \frac{1}{|\boldsymbol{X}^{in}_{b_i}|} \sum_{\boldsymbol{x} \in \boldsymbol{X}^{in}_{b_i}} \mathcal{P}(\text{HSO}|\boldsymbol{x}) - \frac{1}{|\boldsymbol{X}^{ex}_{b_i}|} \sum_{\boldsymbol{x} \in \boldsymbol{X}^{ex}_{b_i}} \mathcal{P}(\text{HSO}|\boldsymbol{x}) \end{aligned} \tag{8}$$

where $\boldsymbol{X}^{in}_{b_i}, \boldsymbol{X}^{ex}_{b_i}$ are the inner and exterior area respectively splited by edge $b_i$.

$\mathcal{P}_{\boldsymbol{\mathcal{E}}}(\boldsymbol{X}_{b_i}|b_i)$ can be estimated using the contrast of the two sides of patch $\boldsymbol{X}_{b_i}$ separated by $b_i$. The higher the contrast, the higher the possibility that $b_i$ is the edge/boundary.

$$\mathcal{P}_{\boldsymbol{\mathcal{E}}}(\boldsymbol{X}_{b_i}|b_i) = exp(\frac{d_{\boldsymbol{C}}(\boldsymbol{X}_{b_i})}{\sigma_{\boldsymbol{C}}} + \frac{d_{\boldsymbol{G}}(\boldsymbol{X}_{b_i})}{\sigma_{\boldsymbol{G}}}) \tag{9}$$

where $d_{\boldsymbol{C}}(\boldsymbol{X}_{b_i}), d_{\boldsymbol{G}}(\boldsymbol{X}_{b_i})$ are the color difference and gradient difference between inner and exterior area centered on $b_i$.

After substituting the equations (4) and (6) into equation

(3), we could get

$$\begin{aligned}\mathcal{B}_{\mathrm{MAP}}(\boldsymbol{X}) &= \arg\max_{\mathcal{B}} \prod_{i=1,2...,W-1} \mathcal{P}(\boldsymbol{X}_{b_i}|b_i)\mathcal{P}(b_i|b_{i-1}) \\ &= \arg\max_{\mathcal{B}} \sum_{i=1,2...,W-1} log[\mathcal{P}(\boldsymbol{X}_{b_i}|b_i)\mathcal{P}(b_i|b_{i-1})]\end{aligned} \tag{10}$$

## 4. DYNAMIC PROGRAMMING IN BOUNDARY SEARCH

In this section, we will use a dynamic programming method to solve the maximization problem of equation (10).

As mentioned before, a grid graph of $W \times H$ nodes is built on the pixels as shown in Figure 3, where each node $v_{i,j}$ represents a pixel in the image. Here we suppose that each node $v_{i,j}$ only has several edges pointing to the adjacent pixels $v_{i+1,j+k}$ on the right with k step(s) difference in y axis. The weight $w(i,j,k)$ of an edge $b_i$ connecting nodes $v_{i,j}$ and $v_{i,j+k}$is defined as the Logarithm of boundary likelihood: $w(i,j,k) = log[\mathcal{P}(\boldsymbol{X}_{b_i}|b_i^{j,k})\mathcal{P}(b_i^{j,k}|b_{i-1})]$

To simplify the formulation, two dummy nodes s and t are added to the above said graph, where s has H outgoing edges connecting to $v_{1,j}$ for $j = 1...H$, and t has H incoming edges connecting from $v_{W,j}$ for $j = 1, ... \cdots H$.

Then, the maximum-weight path from s to t on the graph corresponds to the maximum likelihood estimation of the foreground object boundary. The aforementioned graph is a directed acyclic graph which has no loops, the maximum-weight path on this particular class of graphs can be found efficiently by solving an equivalent maximum-weight path problem on the same graph but with negative weights. This optimization problem can also be formulated as a dynamic program.

Let $f(i,j)$ be the solution to a subproblem of the maximum-weight path from $s$ to $v_{i,j}$. It can be recursively defined by

$$f(i,j) = \begin{cases} 0 & \text{if } i = 1 \\ \max\limits_{k}\{f(i-1,j-k) + w(i,j,k)\} & \text{if } i = 2,...,W \end{cases} \tag{11}$$

and the global optimal solution is $\max\limits_{j} f(W,j)$, while the corresponding optimal path can be calculated by backtracking is the HSO boundary we want.

Using dynamic programming, the value of a maximum-weight path can be computed in $O(W \times H \times K)$ time where K is the number of possible step differences of radius.

## 5. EXPERIMENTAL RESULTS

To evaluate the performance of our proposed HSO segmentation method, extensive experiments are done in comparison with other segmentation algorithms. Some of the resutls are listed in this section. Images shown in Figure 4 represent the results of our proposed HSO segmentation method compared with GrabCut in OneCut algorithm [10]. We could see that GrabCut could have good results under simple background (the first row in Figure 4). When it comes to complex backgrounds, GrabCut algorithm may produce bad results, while our method does not suffer from this because we have taken advantage of the prior knowledge of HSO.

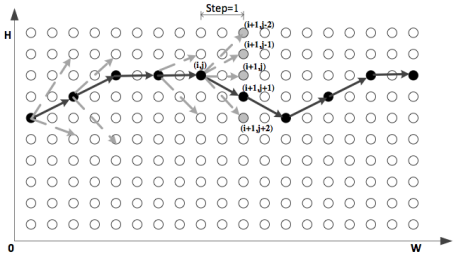


**Fig. 3**: Grid graph built on pixels

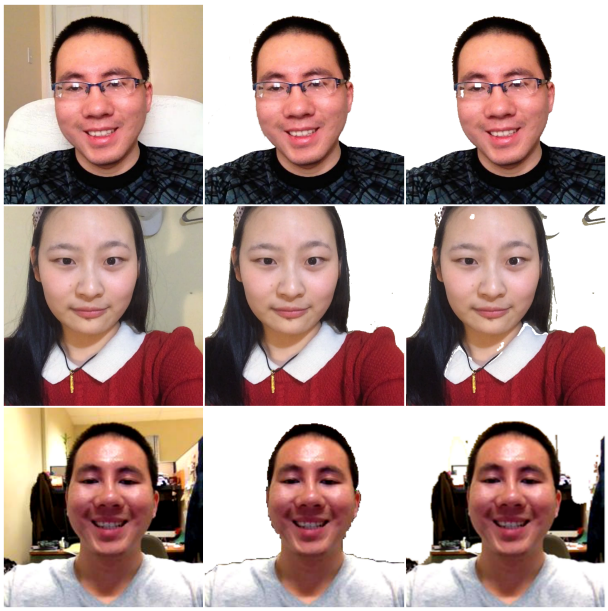

**Fig. 4**: From left to right are input images, segmentation results using our method, and segmentation results using One-Cut method [10].

## 6. CONCLUSION

In this paper, we have proposed a fast head-and-shoulder segmentation method which is capable of extracting frontal human head portraits from arbitrary unknown background. This new method consists of three main phases; 1. modeling human body in polar coordinates; 2. formulating the segmentation problem using Markov chain model; 3. dynamic programming solution to the segmentation problem.

# References


[1] Bishop, C. M. (2006). *Pattern recognition and machine learning*. springer.

[2] Boykov, Y., Veksler, O., and Zabih, R. (2001). Fast approximate energy minimization via graph cuts. *PAMI, IEEE Trans. on*, **23**(11), 1222–1239.

[3] Brice, C. R. and Fennema, C. L. (1970). Scene analysis using regions. *Artificial intelligence*, **1**(3), 205–226.

[4] Cremers, D., Rousson, M., and Deriche, R. (2007). A review of statistical approaches to level set segmentation: integrating color, texture, motion and shape. *International journal of computer vision*, **72**(2), 195–215.

[5] Felzenszwalb, P. F. and Huttenlocher, D. P. (2004). Efficient graph-based image segmentation. *International Journal of Computer Vision*, **59**(2), 167–181.

[6] Horowitz, S. L. and Pavlidis, T. (1976). Picture segmentation by a tree traversal algorithm. *Journal of the ACM (JACM)*, **23**(2), 368–388.

[7] Pavlidis, T. and Liow, Y.-T. (1990). Integrating region growing and edge detection. *PAMI, IEEE Trans. on*, **12**(3), 225–233.

[8] Shi, J. and Malik, J. (2000). Normalized cuts and image segmentation. *PAMI, IEEE Trans. on*, **22**(8), 888–905.

[9] Sun, J., Zhang, W., Tang, X., and Shum, H.-Y. (2006). Background cut. European Conference on Computer Vision. Springer Berlin Heidelberg, 2006

[10] Tang, M., Gorelick, L., Veksler, O., and Boykov, Y. (2013). Grabcut in one cut. Proceedings of the IEEE International Conference on Computer Vision. 2013.

[11] Luo, Huitao, and Alexandros Eleftheriadis. "Model-based segmentation and tracking of head-and-shoulder video objects for real time multimedia services." IEEE Trans. on multimedia 5.3 (2003): 379-389.