EUROGRAPHICS 2012 / P. Cignoni, T. Ertl
(Guest Editors)

Volume 31 (2012), Number 2

Creating Picture Legends for Group Photos

Junhong Gao[1], Seon Joo Kim[2], Michael S. Brown[1]

[1]School of Computing, National University of Singapore
[2]Department of Computer Science, SUNY Korea

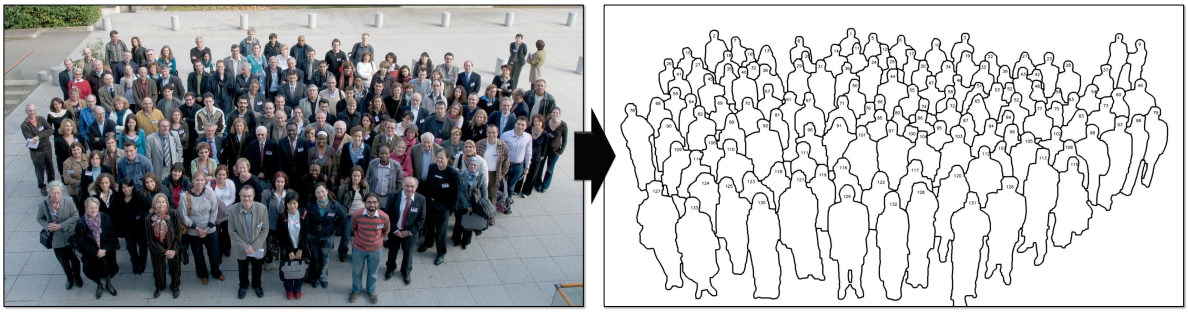

Figure 1: *A group photo of 133 people and the corresponding picture legend created by our approach.*

Abstract

Group photos are one of the most common types of digital images found in personal image collections and on social networks. One typical post-processing task for group photos is to produce a key or legend to identify the people in the photo. This is most often done using simple bounding boxes. A more professional approach is to create a picture legend that uses either a full or partial silhouette to identify the individuals. This paper introduces an efficient method for producing picture legends for group photos. Our approach combines face detection with human shape priors into an interactive selection framework to allow users to quickly segment the individuals in a group photo. Our results are better than those obtained by general selection tools and can be produced in a fraction of the time.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation

1. Introduction

Group photos are taken at a variety of social events from weddings to school and sports events to casual gatherings of friends and family. A common task for group photos, especially those shared on social networks, is to annotate the individuals in the photo. Current approaches found on popular social networks, such as Facebook, use bounding boxes positioned over a person's face. This approach can be ineffective and confusing as multiple faces may be present in a single box. Moreover, when viewing a labeled photo, the user must move the mouse close to the face to see the person's identity; other parts of the body cannot be selected.

A more professional annotation strategy involves creating a *picture legend* as shown in Figure 1. A picture legend [†] is a silhouette (or partial silhouette) of each individual person in the photo. In the printing industry, a picture legend is typically shown below its corresponding photograph and is annotated with numbers together with an associated name list. For digital viewing, the picture legend can be used as a picture map that responds to direct manipulation using a mouse or touch screen. While existing generic image selection methods (e.g. [RKB04,LSS09]) make it possible to seg-

[†] Also referred to as a *group photokey* or *photomap*.

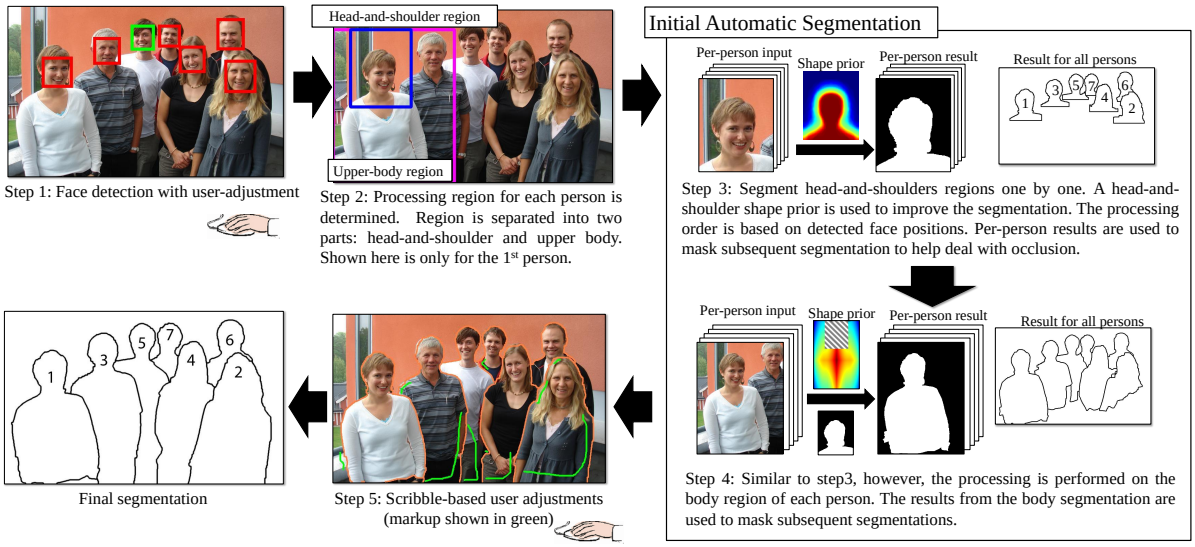

Figure 2: *Overview of our processing pipeline consisting of five steps: 1) face detection with user-adjustment; 2) person ordering and region assignment; 3) segmentation of the head-and-shoulder regions; 4) segmentation of the body regions; 5) user-assisted touch up.*

ment individuals from images, segmenting a large number of individuals in a group photo is time consuming and cumbersome.

To create picture legends in an effective manner, we introduce an interactive framework for human body segmentation which is based on face detection, heuristics on the ordering of faces, and body shape priors. With our system, a user can produce better picture legends more quickly, with significantly less interaction than that required by existing selection tools. We demonstrate the effectiveness of our approach on a variety of examples.

2. Related Work

Our approach combines elements from face/human detection and interactive segmentation. A full discourse on these topics is outside the scope of this paper, here we mention representative works relevant to our paper.

Automatic Human and Face Detection. One of the most well studied and mature methods for detecting the presence of people in a photo is face detection (for a survey see [VJ04]). While face detection is not sufficient for extracting an entire body from an image, it provides significant information regarding the number of people in a photo and their locations. Notable detectors include the work in [VJ01] and [LM02], which are robust and also available as APIs.

There are also approaches that focus on detecting the whole body. Successful methods typically combine one or more low level features such as histogram of oriented gradient [DT05, ZYCA06] with various analysis strategies [TPM07, LD08, WN09, SKHD09] to detect the presence and the locations of humans in an image. These approaches are often more concerned with detecting the presence of humans rather than the segmentation.

There are also a body of work on human pose estimation, in which the interest is in extracting more localized spatial information compared to face and body detection. Such methods usually decompose a detected human body into sub-components and either register them individually [BM09, BMBM10] or use a hinge model to reduce the feasible space of human poses [Ram06, FMJZ08, ARS09, YL10]. These approaches, however, are still not accurate enough for pixel-level segmentation and are not designed for cluttered scenes with significant occlusions as shown in Figure 1.

There is also recent work that build upon the ability to detect faces and bodies in group photos to determine high-level semantics such as gender or social relationships of members in the photos [GC09, WGLF10], or to assist in the pose estimation [EF10, PESG09]. However, such group photo analysis has yet been used to assist in the segmentation of the individuals.

Work most closely related to ours are those aimed at automatically extracting the contours of humans from images [FWZB10, GFB10]. As with the pose detection, these approaches are not well suited for handling inputs with significant occlusions. As far as we are aware, there are no existing approaches that can automatically segment a group photo to the level required to produce a good picture legend.

Interactive Image Segmentation. Interactive image segmentation can be categorized as scribble-based, painting-

based, or boundary-based. Scribble-based approaches [LSkTyS04, BVZ01, RKB04] segment objects using a series of user drawn scribbles to denote regions of the object's foreground and areas of background. These scribbles provide training data that allows these approaches to classify unmarked pixels to either the foreground or the background classes.

Painting-based segmentation [LSS09] allows the user to interactively segment an object by painting on the foreground object. These approaches do not need to explicitly denote the background pixels; background and foreground training data are estimated on the fly from the unknown region. Boundary-based approaches [MB95, KWT88] do not explicitly label pixels as foreground, but instead define a closed boundary by tracing salient edge structures.

It is possible to produce a picture legend using these generic interactive segmentation methods, however, it is time consuming and tedious as these approaches are not designed to perform multiple object segmentation. As a result, each person has to be segmented one by one. A closely related work to ours is that of RepFinder [CZM*10] which exploits the similarity in geometric shape and colors to find repetitive objects. Our approach is similar in nature, however, RepFinder targets objects that are nearly identical, whereas individuals in a group photo exhibit too great of a diversity due to variations in their hair, face, clothing and pose.

3. Overall Procedure

Figure 2 illustrates our overall procedure which is divided into five steps. Step 1 performs an initial face detection to find the location of each person in the image. If necessary, the user can add/delete faces or make adjustments to the face locations. In step 2, the processing region for each person is defined. Regions are separated into two subregions: head-and-shoulder and body regions. Overall, the regions for all people in the image are ordered based on the detected face locations in a bottom up fashion. Steps 3 and 4 perform an initial segmentation using a graph-cut optimization. This is done separately for the head-and-shoulder and the body region. Shape priors corresponding to these regions are incorporated to help improve the results. This initial segmentation is performed automatically, one by one, based on the individual's spatial ordering. The results from segmented regions are used as masks for subsequent segmentations, e.g. pixels assigned to person i are not considered when segmenting person $i+1$. Finally, step 5 provides the user with a scribble-based interface to touch up the initial segmentation. Details for each of these steps are discussed in the following sections.

4. Face Detection and Process Region Assignment

We perform face detection using the OpenCV API [Bra00] which is based on methods in [VJ01, LM02]. This procedure returns the location of the face as a rectangle, represented as (x, y, w, h), where x, y is the center of the face and w, h is the width and height of the rectangle. To compensate for any errors, the user can iteratively adjust the position of the detected faces, add missing faces, or delete erroneously detected faces. The goal now is to segment the entire image such that each pixel in the image is assigned to a label, $l \in \mathcal{L}_U = \{l_0, l_1, ..., l_n\}$ where n is the total number of people in the photo, l_0 is the label for the background and l_i is the ith person.

Next, we assign a processing region to each of the n individuals based on their face locations. Each ith person's region is divided into two parts: a head-and-shoulder region and a body region as shown in Figure 2. The size of the body regions are heuristically defined as $4w$ and $6h$, while the head-and-shoulder region is $2w$ and $2.5h$, with the additional $0.5h$ added to the bottom to compensate for the shoulders.

For each ith person, the processing regions define the space where segmentation will be performed. Moreover, the subregions of the head-and-shoulder and body are segmented separately, i.e. the head-and-shoulder of a person i is segmented only within the head-and-shoulder region, while the body is segmented only within the body region, exclusive of the head-and-shoulder region. Our scheme segments the head-and-shoulder for each person first, one by one, from bottom of the image to top, based on the detected face spatial positions (x, y). After all n head-and-shoulder results are segmented, the associated bodies are segmented in the same order.

Segmentation is performed in this manner, because a sig-

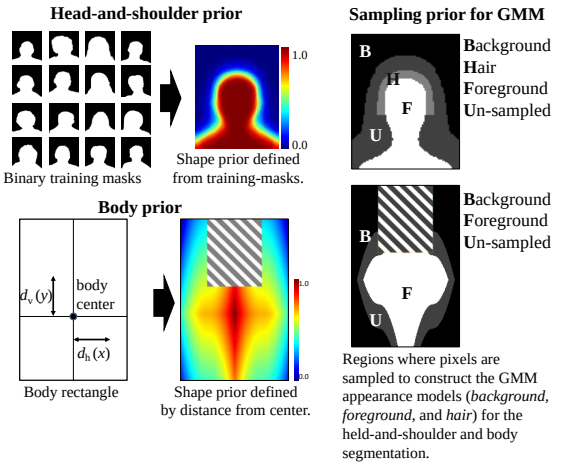

Figure 4: *(Left) This shows how the head-and-shoulder and body shape priors are defined. (Right) This shows how the pixels are sampled to produce the GMM appearance models to be used for the head-and-shoulder and body segmentation.*

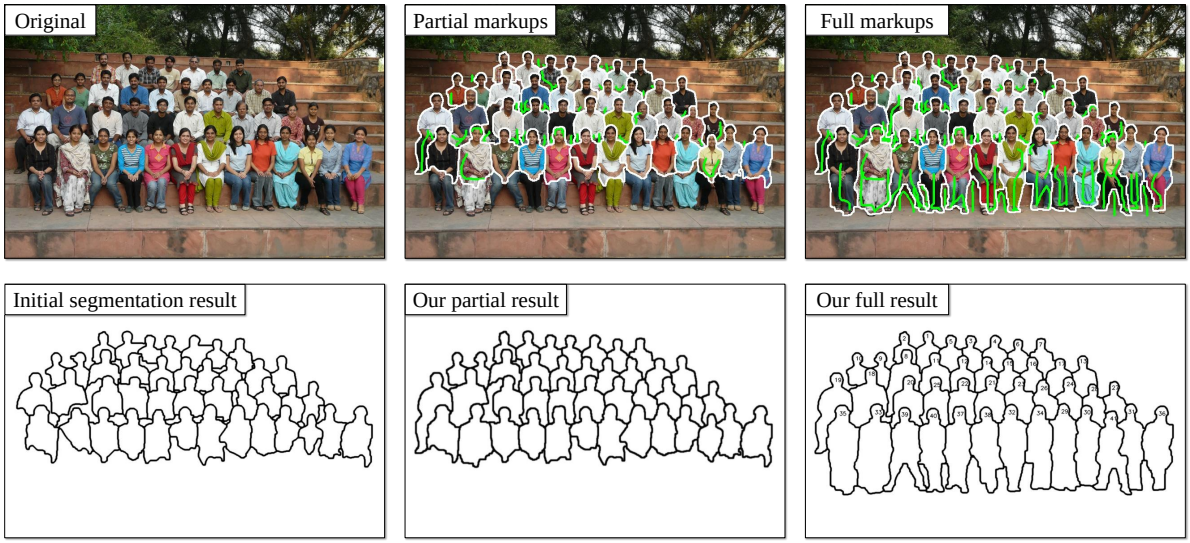

Figure 3: *This figure shows the initial segmentation of our algorithm followed by two different markups. We first show the markup (shown in Green) required to obtain partial silhouettes of the persons in the photo. Such partial results are sufficient for many applications including for printing and online annotation. The second results shows markup made to perform full silhouettes of each individual.*

nificant portion of the head-and-shoulders regions are guaranteed to appear in the image for all individuals. We segment starting from the bottom of the image because the people in front are more likely to have their whole body shown in the image compared to the people in the back who will have most of their bodies occluded by other peoples' faces and bodies.

5. Segmentation Priors

5.1. Shape Priors

We introduce two shape priors that assist the segmentation of the head-and-shoulder and body regions. Since faces in group photos are fairly regular and in an upright position, we compute a shape prior from a training data set from over a 100 examples of segmented head and shoulders that have been cropped from different group photos (see supplemental material). The head-and-shoulder prior map, M_h, is computed from the binary training images (I_i) as follows:

$$M_h(x,y) = \frac{\sum_i^N I_i(x,y) + \sum_i^N I_i^f(x,y)}{2N}, \qquad (1)$$

where I_i^f indicates a horizontally flipped image. Figure 4 shows examples of the segmented training data and the resulting M_h that encodes each pixel's probability that it belongs to the foreground.

Designing a similar body prior map is more difficult due to the variety of poses that a human body can assume. While the head will be mostly upright in group photos, body pose can vary from person to person and from image to image. Therefore, we use a simple heuristic that is based on the distance to the body center [‡] to construct the body prior map as follows:

$$M_b(x,y) = (1 - d_v(y)) \cdot (1 - d_h(x))^2 \qquad (2)$$

where $d_h(\cdot)$ and $d_v(\cdot)$ are the normalized horizontal and vertical distance functions, i.e. $d_v(\cdot) \in [0,1]$ and $d_h(\cdot) \in [0,1]$, from the center of the body rectangle to the edges. Note that the horizontal distance $d_h(x)$ is squared to slightly reduce the influence in the horizontal direction. The body prior map is shown in Figure 4.

5.2. Segmentation via Graph Cut

While the final segmentation result is a multi-label output, we treat the segmentation of each ith person as a binary labeling problem, formulated using a Markov Random Field as typically done in interactive segmentation algorithms [LSkTyS04, RKB04, LSS09]. In particular, each pixel $\mathbf{p}$ in the processed region is assigned a label $l \in \{0,1\}$, representing *background* or *foreground* respectively. This segmentation is done by minimizing the following energy function:

$$E = \sum_{\mathbf{p}} E_d(l_{\mathbf{p}}) + \sum_{(\mathbf{p},\mathbf{q}) \in \mathbf{N}} \lambda E_s(l_{\mathbf{p}}, l_{\mathbf{q}}), \qquad (3)$$

[‡] The center of the body is offset by the $0.5h$ in the vertical direction, where h is the size of the initial detected face.

where E_d is the data-cost associated with assigning a pixel to a particular label, and E_s is the smoothness cost associated with label assignments between pixels $\mathbf{p}$ and $\mathbf{q}$ which belongs to 4-connected neighborhood set $\mathbf{N}$.

As previously mentioned, segmentation is first applied to the head-and-shoulder region and then to the body region. To perform the head-and-shoulder segmentation, we build three different appearance models: *foreground*, *background*, and *hair*. The appearance model for the foreground, $p^f(\cdot)$, the background $p^b(\cdot)$, and the hair, $p^h(\cdot)$, are computed by fitting Gaussian Mixture Models (GMM) [RKB04] with four components for $p^f(\cdot)$ and $p^b(\cdot)$ and two components for $p^h(\cdot)$. The sampling regions where the GMM training samples (i.e. RGB pixels) are selected from are shown in Figure 4. Appendix A provides details on how to construct these sampling regions.

The data-cost, E_d, for the head-and-shoulder segmentation is defined as:

$$E_d(l_\mathbf{p}=1)=\begin{cases}\infty & \text{if} \quad M_h(\mathbf{p})=0\\ \min(L^f_\mathbf{p},L^h_\mathbf{p})\cdot(1-M_h(\mathbf{p})) & \text{otherwise}\end{cases}$$

$$E_d(l_\mathbf{p}=0)=\begin{cases}\infty & \text{if} \quad M_h(\mathbf{p})=1\\ L^b_\mathbf{p}\cdot(M_h(\mathbf{p})) & \text{otherwise}\end{cases}$$

where M_h is the head-and-shoulder prior map defined in Section 4, and $L^*_\mathbf{p}=-\ln p^*(I_\mathbf{p})$ is the logarithm of the probability of GMM $(*)$ for the three appearance models.

The data-cost for the body segmentation is defined in a similar fashion:

$$E_d(l_\mathbf{p}=1)=\begin{cases}\infty & \text{if} \quad M_b(\mathbf{p})=0\\ L^f_\mathbf{p}\cdot(1-M_b(\mathbf{p})) & \text{otherwise,}\end{cases}$$

$$E_d(l_\mathbf{p}=0)=\begin{cases}\infty & \text{if} \quad M_b(\mathbf{p})=1\\ L^b_\mathbf{p}\cdot(M_b(\mathbf{p})) & \text{otherwise,}\end{cases} \tag{4}$$

where M_b is the body prior map defined in Section 4, and the $L^f\mathbf{p}$ and $L^b\mathbf{p}$ are defined using GMM appearance models sampled from the body region in the areas shown in Figure 4.

The smoothness term for both the head-and-shoulder segmentation and body segmentation is defined the same as:

$$E_s(l_\mathbf{p},l_\mathbf{q})=|l_\mathbf{p}-l_\mathbf{q}|\cdot(1+\|I_\mathbf{p}-I_\mathbf{q}\|^2)^\beta, \tag{5}$$

where the scaling exponent β is set to -0.8 constantly.

After each segmentation, the label id i of the current person being processed is assigned to all pixels labeled as *foreground*. Note that this binary segmentation does not include any pixels that have already been assigned a label from prior segmentations, i.e. pixels already labeled with person id i-1 (or less) are not processed in subsequent segmentations. An example of the result obtained by this automatic segmentation is shown in Figure 3.

6. Segmentation Touch up

The results from our automatic segmentation typically require touch ups. This is performed using a scribble based approach where the user draws foreground strokes to specify corrections. Recall that after our initial segmentation, each pixel in the image belongs to one of $n+1$ labels, where l_i represents the ith person and l_0 represents the background.

The correction is performed as a binary labeling similar to the initial segmentation described in Section 5.2. To perform a correction, the user draws a scribble from one labeled region to another. For example, if the user wants to correct a problem with person i, they draw a scribble starting in a region labeled with l_i into another region that has label l_j. The correction is applied only to pixels that fall within the two regions touched by the scribble as shown in Figure 5(a).

Once the scribble is drawn for person i, our approach treats all pixels with label l_i as definite foreground, as well as the pixels marked by the scribble. All other pixels with label l_j are treated as the unknown region (see Figure 5(b)). For the appearance model in this touch up segmentation, we use the ith person's GMM model as the foreground appearance, and the jth person's GMM model to model the background. The smoothness cost is the same as in Section 5.2.

We again use the graph-cut segmentation method which was described in Section 5.2 to compute the updated result. The running time for each update is very fast (usually around 0.1sec). After this new binary segmentation is completed, all foreground labels are assign the value l_i. Note that while we have described this procedure for markup that only covered two regions, i.e. i and j, this correction procedure easily extends when the scribble crosses multiple labels. In such cases, we treat all pixels from region i and user scribble as foreground, and all pixels from the other labels as the unknown region.

We also note that the exact same procedure is used when the user scribble starts in the background region, l_0. In this situation, we treat l_0 as the foreground class, and l_j as the unknown region. The final segmentation result will assign all foreground results to l_0.

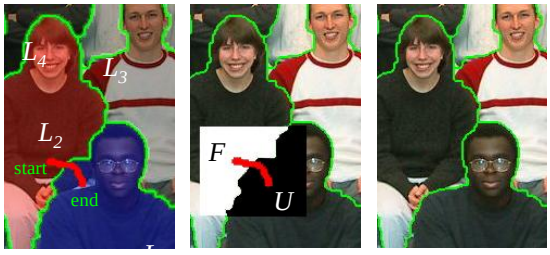

Figure 5: *Example of scribble-based touch up. (a) User's scribble (red). (b) Localized box about the markup. (c) Final result.*

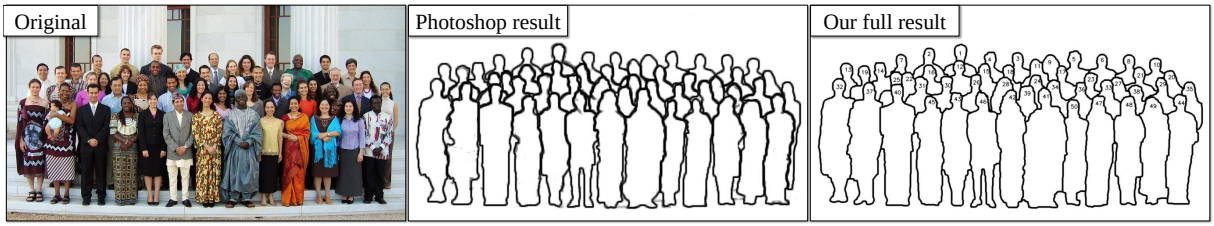

Figure 6: *A comparison of the results using our technique and Photoshop [Pho]. (a) Original image. (b) Segmented result obtained using Photoshop took around 30 minutes. (c) Our fully segmented result which took less than 5 minutes.*

7. Results

Figures 3, 6, 7 show results generated using our approach on a variety of input images. Our images are of various input sizes, however, to provide interactive speeds, processing is performed on down-sampled versions approximately 800×600 in resolution, the results of which are upsampled back to the native resolution. The number of people in these images range from 13 to 50.

Figure 3 shows a group photo with 41 people. The typical markup needed to perform the touch up of our initial segmentation is shown in green. We show the markups needed to produce picture legends for both the partial and the full silhouettes. While most of the results shown in this paper are full silhouettes, we note that partial silhouettes are likely to be sufficient for many applications.

Figure 6 shows a comparison of our results with one obtained using Abode Photoshop. The Photoshop results were obtained by a user familiar with Photoshop. The user segmented each individual in the photo one at a time using the quick selection tool [LSS09] provided by Photoshop. After an individual was segmented, a mask was assigned to the pixels to prevent them from being selected in subsequent segmentations. Our results took roughly 5 minutes to obtain, while the Photoshop result took nearly 30 minutes. In addition, the Photoshop result suffers from uneven borders in the final output due to edges of adjacent individuals being outlined twice. We note that we perform a true segmentation for the Photoshop result, i.e. each person has an associated mask. This segmentation approach is more time consuming than simply drawing curves in an freehand manner to trace the silhouettes. However, unlike a true segmentation, traced curves are not suitable to be used in interactive applications that enable users to click on individuals in the group photo.

	no. of per	Est.	Partial	Full
Figure 1	133	72s	8min41s	10min33s
Figure 3	41	22s	1min43s	2min43s
Figure 6	50	31s	3min40s	4min15s
Figure 7(a)	48	35s	2min07s	3min38s
Figure 7(b)	18	21s	1min32s	2min13s
Figure 7(c)	29	22s	2min32s	3min08s
Figure 7(d)	13	35s	1min17s	2min21s
Figure 7(e)	23	18s	2min10s	3min06s
Figure 7(f)	19	16s	1min44s	3min00s
Figure 7(g)	20	33s	1min40s	2min40s

Table 1: *Processing times for results shown in this paper. The first column shows the number of individuals in the group photo. The second column lists processing time to perform the automatic segmentation. The third and forth columns show the interactive time needed to produce the partial and full silhouette results.*

Several more examples are presented in Figure 7. Table 1 shows the times needed to produce the automatic segmentation results and the user interaction time to produce the partial and full silhouette results. The only other result we attempted with Photoshop was Figure 7(a) which took approximately 18 minutes.

8. Discussion

We have presented an approach to create picture legends from group photos. Our approach represents strategies that were found to be effective for performing this task. For example, we initially formulated the problem as a multilabel MRF (i.e. a label for each person), but found the optimization time to be too slow. As a result, we opted to perform binary segmentation in a one-by-one manner. This strategy worked well at handling occlusion, especially when combined with region ordering and the separation of the head-and-shoulder and body region. We also found our simple scribble based touch up to be effective. This method provides a functionality very similar to paint selection [LSS09] but does so in a multi-label environment. The multilabel nature of our results provides results with uniform borders that is difficult to achieve in Photoshop.

We also found that our simple head-and-shoulder and body prior worked well in assisting the initial automated segmentation. This is because individuals in most group photos stand upright in a reasonable ordered fashion as seen in

Figure 7: *Images (a-g) show various input processed by our approach to obtain picture legends with full silhouettes. The number of people in the images ranges from 13 to 50.*

our various examples (Figures 3, 6, 7). However, our human body priors will have difficulties with uncommon poses and arm movements since our model does not account for such cases (Figure 8).

In the future, we plan to incorporate body priors based on a robust pose analysis such as in [FWZB10, EF10]. Moreover, there is still room for improvement in analyzing the ordering of people. While the current bottom-to-top approach handles normal poses, a more sophisticated method will be necessary to deal with various poses which makes the topology of the group more complicated.

Acknowledgement

This work was supported by the Singapore Academic Research Fund (AcRF) Tier 1 FRC grant (Project No. R-252-000-423-112).

References

[ARS09] ANDRILUKA M., ROTH S., SCHIELE B.: Pictorial structures revisited: People detection and articulated pose estimation. In *CVPR* (2009).

[BM09] BOURDEV L., MALIK J.: Poselets: Body part detectors trained using 3d human pose annotations. In *ICCV* (2009).

[BMBM10] BOURDEV L., MAJI S., BROX T., MALIK J.: Detecting people using mutually consistent poselet activations. In *ECCV* (2010).

[Bra00] BRADSKI G.: The OpenCV Library. *Dr. Dobb's Journal of Software Tools* (2000).

[BVZ01] BOYKOV Y., VEKSLER O., ZABIH R.: Fast approximate energy minimization via graph cuts. *TPAMI 23* (2001), 1222–1239.

[CZM*10] CHENG M., ZHANG F., MITRA N., HUANG X., HU

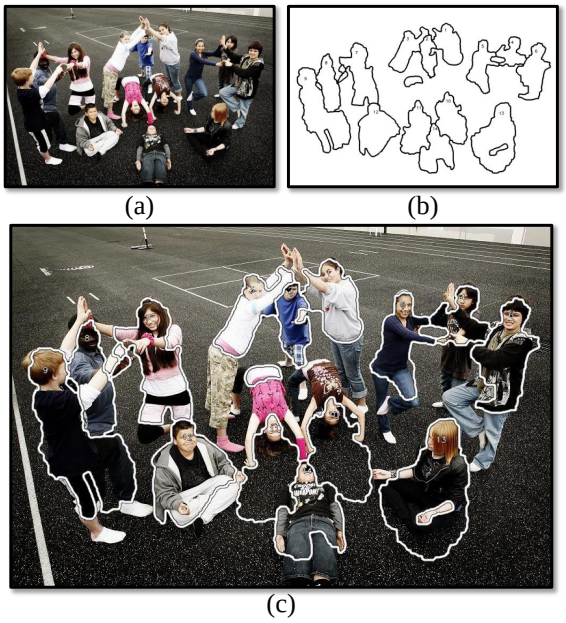

Figure 8: *A difficult case for our method to handle. (a) Original image. (b) Initial silhouette estimations. (c) Segmentation boundary superimposed on the image.*

S.: Repfinder: Finding approximately repeated scene elements for image editing. *ACM ToG 29*, 3 (2010).

[DT05] DALAL N., TRIGGS B.: Histograms of oriented gradients for human detection. In *CVPR* (2005).

[EF10] EICHNER M., FERRARI V.: We are family: Joint pose estimation of multiple persons. In *ECCV* (2010).

[FMJZ08] FERRARI V., MARÍN-JIMÉNEZ M. J., ZISSERMAN A.: Progressive search space reduction for human pose estimation. In *CVPR* (2008).

[FWZB10] FREIFELD O., WEISS A., ZUFFI S., BLACK M. J.: Contour people: A parameterized model of 2d articulated human shape. In *CVPR* (2010).

[GC09] GALLAGHER A. C., CHEN T.: Understanding images of groups of people. In *CVPR* (2009).

[GFB10] GUAN P., FREIFELD O., BLACK M. J.: A 2d human body model dressed in eigen clothing. In *ECCV* (2010).

[KWT88] KASS M., WITKIN A. P., TERZOPOULOS D.: Snakes: Active contour models. *IJCV 1*, 4 (1988), 321–331.

[LD08] LIN Z., DAVIS L. S.: A pose-invariant descriptor for human detection and segmentation. In *ECCV* (2008).

[LM02] LIENHART R., MAYDT J.: An extended set of haar-like features for rapid object detection. In *ICIP* (2002).

[LSkTyS04] LI Y., SUN J., KEUNG TANG C., YEUNG SHUM H.: Lazy snapping. *ACM ToG 23* (2004), 303–308.

[LSS09] LIU J., SUN J., SHUM H.: Paint selection. *ACM ToG 69* (2009), 1–7.

[MB95] MORTENSEN E. N., BARRETT W. A.: Intelligent scissors for image composition. In *SIGGRAPH* (1995).

[PESG09] PELLEGRINI S., ESS A., SCHINDLER K., GOOL L. J. V.: You'll never walk alone: Modeling social behavior for multi-target tracking. In *ICCV* (2009).

[Pho] PHOTOSHOP:. Adobe Inc., Adobe Photoshop CS5, http://www.adobe.com/products/photoshop.

[Ram06] RAMANAN D.: Learning to parse images of articulated bodies. *NIPS* (2006).

[RKB04] ROTHER C., KOLMOGOROV V., BLAKE A.: "grabcut": Interactive foreground extraction using iterated graph cuts. *ACM ToG* (2004), 309–314.

[SKHD09] SCHWARTZ W. R., KEMBHAVI A., HARWOOD D., DAVIS L. S.: Human detection using partial least squares analysis. In *ICCV* (2009).

[TPM07] TUZEL O., PORIKLI F., MEER P.: Human detection via classification on riemannian manifolds. In *CVPR* (2007).

[VJ01] VIOLA P. A., JONES M. J.: Rapid object detection using a boosted cascade of simple features. In *CVPR* (2001).

[VJ04] VIOLA P., JONES M.: Robust real-time face detection. *IJCV 57* (2004), 137–154.

[WGLF10] WANG G., GALLAGHER A. C., LUO J., FORSYTH D. A.: Seeing people in social context: Recognizing people and social relationships. In *ECCV* (2010).

[WN09] WU B., NEVATIA R.: Detection and segmentation of multiple, partially occluded objects by grouping, merging, assigning part detection responses. *IJCV 82*, 2 (2009), 185–204.

[YL10] YAO B., LI F.-F.: Modeling mutual context of object and human pose in human-object interaction activities. In *CVPR* (2010).

[ZYCA06] ZHU Q., YEH M., CHENG K., AVIDAN S.: Fast human detection using a cascade of histograms of oriented gradients. In *CVPR* (2006).

Appendix A: Definition of Sampling Regions

The size of the head-and-shoulder and body regions were defined after investigating the human body ratio from a large amount of human figures in group photo. For each individual we have the detected face rectangle x, y, w, h, where x, y is the center of the face and w, h are the width and height. The bounding rectangular expressed as the upper left and lower right points for the head-and-shoulder region, R_h is $(x-w, y-1.5h, 2w, 2.5h)$. The body region R_b is defined as $(x-2w, y-3h, 4w, 6h)$.

The head-and-shoulder sampling regions are computed based on the head-and-shoulder prior, M_h, and are defined as:

$$\begin{array}{l} \textbf{Background} : \{(x,y) \mid M_h(x,y) \leq 0.1\} \\ \textbf{Foreground} \ : \{(x,y) \mid M_h(x,y) \geq 0.9\} \\ \textbf{Hair} : \{(x,y) \mid 0.5 \leq M_h(x,y) \leq 0.9 \text{ and } y > 0.6h\} \end{array} . \quad (6)$$

All other points are consider to be in the unknown region.

The body sampling regions are defined from the body prior, M_b, as:

$$\begin{array}{l} \textbf{Background} : \{(x,y) \mid M_b(x,y) \leq 0.4\} \\ \textbf{Foreground} \ : \{(x,y) \mid M_b(x,y) \geq 0.3\} \end{array} . \quad (7)$$

All other points are consider to be in the unknown region.