PAPER

Using Super-Pixels and Human Probability Map for Automatic Human Subject Segmentation

Esmaeil POURJAM[†a)], *Nonmember*, **Daisuke DEGUCHI**[††], *Member*, **Ichiro IDE**[†], *Senior Member*, *and* **Hiroshi MURASE**[†], *Fellow*

SUMMARY Human body segmentation has many applications in a wide variety of image processing tasks, from intelligent vehicles to entertainment. A substantial amount of research has been done in the field of segmentation and it is still one of the active research areas, resulting in introduction of many innovative methods in literature. Still, until today, a method that can overcome the human segmentation problems and adapt itself to different kinds of situations, has not been introduced. Many of methods today try to use the graph-cut framework to solve the segmentation problem. Although powerful, these methods rely on a distance penalty term (intensity difference or RGB color distance). This term does not always lead to a good separation between two regions. For example, if two regions are close in color, even if they belong to two different objects, they will be grouped together, which is not acceptable. Also, if one object has multiple parts with different colors, e.g. humans wear various clothes with different colors and patterns, each part will be segmented separately. Although this can be overcome by multiple inputs from user, the inherent problem would not be solved. In this paper, we have considered solving the problem by making use of a human probability map, super-pixels and Grab-cut framework. Using this map relives us from the need for matching the model to the actual body, thus helps to improve the segmentation accuracy. As a result, not only the accuracy has improved, it also became comparable to the state-of-the-art interactive methods.

key words: *human segmentation, Grab-cut, super-pixel, human probability map, texture map*

1. Introduction

Human body segmentation is one of useful and attractive areas in the fields of vision which has challenged many researches for a long time resulting in a wide variety of methods and algorithms introduced in literature. Although vastly studied, a method that can perform the segmentation task automatically with good accuracy and adaptivity to different situations is yet to be introduced. The main problem can be said to be the problem with the variability of the human body shape and various combinations of clothes that humans wear leading to a vast amount of combinations of color and texture.

Generally speaking, the segmentation task can be defined in a semi-automatic (interactive) or automatic manner with a generous research done in each field. Usually, automatic segmentation algorithms like those in [1]–[7] are useful for the cases that the numbers of images/subjects to be segmented are not predefined (e.g. driver assistance systems, entertainment systems) or the number is too big to be segmented manually (e.g. image/video archives), while interactive algorithms like those in [8]–[14] are preferred for applications in which the number of images or subjects to be segmented are limited so that the user interaction cost would be feasible (e.g. medical imaging). In these methods, usually, user interaction is tried to be reduced as much as possible.

Many of recent methods try to solve the segmentation problem by utilizing the graph-cut framework. Although powerful, these methods usually rely on a distance penalty term (intensity difference or RGB color distance) which does not always lead to a good separation between regions. For example, even if two regions are similar in color but belong to two different objects, they will be grouped together. The same problem might occur if one object has different parts with several colors, e.g. in case of humans wearing clothes with different colors and patterns, different parts will be segmented separately (As depicted in Fig. 1 top row). Some methods have tried to overcome this problem by adding different types of information or making limitation on the proximity of the object to be segmented. For example, we [3] have proposed that by using shape information, segmentation accuracy can be improved. Likewise, Prakash et al. [7] showed that by using an active contour for finding boundary of the object and using Grab-cut [13] inside, it is possible to improve accuracy. Still, these methods fail to segment an object correctly when the boundary of the object is not well defined or the color of the object and its background are very close to each other.

In present work, we have considered solving the problem of need for user interaction (thus proposing an automatic segmentation system) through using a human body probability map for selecting human body regions from super-pixels generated from an input image. We also tried to overcome the problem of color feature failure due to the changes in the texture of the object (in case of human body, using just color feature usually fails to segment different parts of the body/clothes due to changes in color or texture as in Fig. 2 or Fig. 1 top row), by applying a texture feature in the process of super-pixel generation. The main reason for this selection is 1) Using the probability map relieves us from the need for matching the shape model and actual body, thus helps to improve the segmentation accuracy. 2) Incorporating texture

Manuscript received July 21, 2015.
Manuscript revised December 22, 2015.
[†]The authors are with Graduate School of Information Science, Nagoya University, Nagoya-shi, 464-8601 Japan.
[††]The author is with Information Strategy Office, Nagoya University, Nagoya-shi, 464-8601 Japan.
a) E-mail: esmaeilp@murase.m.is.nagoya-u.ac.jp
DOI: 10.1587/transfun.E99.A.943

information can be done in super-pixels generation, since this kind of information which is not used by the mentioned methods, can improve the accuracy of segmentation regions that color feature alone is not enough. 3) Using the coarse-to-fine or iterative schemes like the one proposed in our previous works [3] or [6], will be taxing on time and computation power, so here, we propose an algorithm to perform the segmentation in one iteration. This results in much less strain on the CPU, thus the process becomes much faster (almost by the scale of 10) while maintaining almost the same accuracy as our previous methods.

The main idea comes from puzzle games. Usually humans wear different clothes with various colors and textures. If we divide the image into regions based on their color/texture, each part of the human body then, becomes like a piece of puzzle. So we can think of an image as a puzzle with multiple pieces in which the human body occupies some of them. If we can select the right pieces, we can have a somewhat rough shape of the body and by using the Grab-cut method, we can segment the human subject accurately. Following this idea, we show that not only the system becomes automatic but also the accuracy of the system improves. We also show that just by using the information of a single image, it is possible to achieve segmentation results with accuracy comparable to the state-of-the-art and much better than traditional methods while having a relatively simpler model. It is also good to note that this method can be used in both automatic and interactive segmentation manners.

The rest of the paper is arranged as follows. In Sect. 2, we present a brief explanation about the related works. Section 3 will explain the method proposed in this work. Section 4 will provide the experimental setup and results of the proposed method compared with other methods. Sections 5 and 6 will respectively belong to discussion and conclusion of the work.

2. Related Works

Many segmentation methods exist in literature, each with their benefits and shortcomings. There also exist surveys like the one by Weinland et al. [15] that explain some famous methods. Here, we briefly introduce some of the methods which are related to our work.

2.1 Interactive Segmentation

Rother et al. [13] in 2004, introduced the Grab-cut segmentation algorithm which is a way for 2D binary labeling (Foreground/Background) in an effective way, using user interaction. For this, a user inputs a polygon around the object-of-interest (The minimum number of points to input is two for defining a box around the object). Based on user selection, two Gaussian Mixture Models (GMM); one for foreground and one for background, are trained. By considering the image pixels as graph nodes and solving max-flow/min-cut in the graph, foreground and background are separated. If needed, the user can input some corrections for achieving

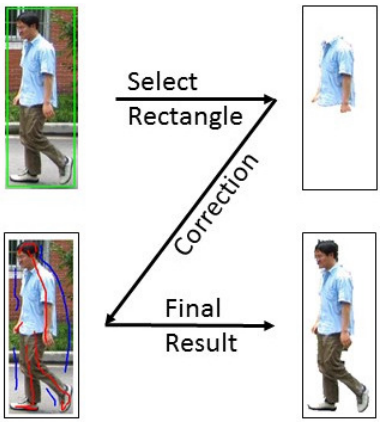


Fig. 1 Example of the segmentation process by the Grab-cut method.

Fig. 2 Grab-cut failing to segment the human subject due to similarity of the color between body parts and background, and also changes in the color of clothing.

better results. The main problem of this method is that the main separation factor is the color distance between two pixels, so if this distance is negligible, the desired result might not be achieved. Also for obtaining a satisfactory result, the user might be forced to perform many corrections. An example of the segmentation process by the Grab-cut method is presented in Fig. 1.

Schmidt et al. [16] proposed a method that can cut the graph faster than other similar graph-cut based methods. They have done this for a subset of graphs called planar graphs and showed that their method can be used for both image segmentation and shape matching. They use the result of the work done by Weihe [17] which shows that the maximal flow in a graph with N vertices can be calculated in $O(N \log N)$ and used the idea to always augment the leftmost of all paths from source to sink in the graph to solve the implementation problems of the original paper, thus achieving computational time of $O(N \log N)$ for image segmentation and $O(N^2 \log N)$ for shape matching.

Gorelick et al. [12] have used a shape convexity prior for interactive image segmentation. Their motivation for this is the significance of the effect of convexity on human vision. Since their proposed prior has non-submodular properties, they utilized dynamic programming and by using length regularization prior alongside a trust region framework, tried to solve the problem. They also proposed that their convexity prior is practically parameter free (no need for parameter optimizations) and can become scale invariant.

Felzenwalb et al. [18] have introduced a segmentation

algorithm which uses their proposed dissimilarity measure (which they call "region comparison predicate") in a graph created from the input image for capturing the conceptually important areas in the image. Their measure uses the *internal difference* of a region in the graph in comparison with the *external difference* of the neighboring regions. For *internal difference* they find the maximum weight of edges of Minimum Spanning Tree (MST) inside the region, while for *external difference*, the minimum weight of the edges connecting two regions is selected. If the *external difference* is less than the minimum of the *internal differences* between two regions, they are merged together. To control the degree of difference between internal and external differences, they use a threshold based on the size of each region. The result will be a segmentation that is not *too coarse* nor *too fine* based on their definitions. They also show that their method can run in near linear time ($O = m \log m$ for m edges in the graph).

By introducing an L_1 distance measure and applying a shape overlap measure into energy minimization model of Grab-cut, Tang et al. [11] have tried to improve the object segmentation accuracy. They replace the concave volume balancing term in Grab-cut formulation with a much simpler L_1 distance measure and then redefine the energy model to adapt to this new parameter. They also have increased the number of color bins used in their work to improve the information acquisition from the input image (they usually use 128^3 bins for color images), which they say can improve the discrimination between an object and its background. Also the new definition of their measure and energy model allows a global minimization in just one graph segmentation step in contrast to the iterative energy minimization algorithm used in Grab-cut.

The name "interactive" indicates that all of the above mentioned methods are completely dependent on user inputs so in applications that user interaction becomes difficult or impossible, they will render useless despite having good accuracy. Also, since all of the mentioned methods use just color feature as the main criteria for segmentation (aside from the work of Felzenwalb et al. [18] that has the potential to incorporate other types of features, although finding the criteria to incorporate them can become tricky), as mentioned before, when the object color and texture changes, they will most probably fail to show good performance.

2.2 Automatic Segmentation

Gulshan et al. [5] have proposed a method for human subject segmentation based on a top-down/bottom-up scheme. For the top-down part, they proposed a Histogram of Oriented Gradients-Support Vector Machine (HOG-SVM) classifier trained using a big training set created by color/depth images taken by Microsoft Kinect. This classifier tries to make a rough estimation of the body in an input image which is a frame with the human subject inside (output of a human detection algorithm) to predict the rough shape of the body. This estimation will then be refined by the bottom-up stage in which a local Grab-cut is used. Their local Grab-cut uses a local color model instead of the global one used by the original Grab-cut by Rother et al. [13]. They claim that using this model improves the discrimination between foreground and background. They also make their method autonomous by incorporating the human detection framework of Felzenwlab et al. [19] which provides the system with the frame around the subject as mentioned above.

Prakash et al. [7] use conventional Active Contours [10] and Grab-cut [13] in parallel for segmenting an object. After user input, the input image is once segmented with active contour to achieve the outside boundary of the object and is once segmented using Grab-cut for extracting the inside objects. By comparing both segmentation results and using a probability fuzzy decision system, they try to make the best decision on each pixel of the image labeling for foreground or background.

Zhang et al. [4] tried to make a robust segmentation system to extract the primary object in videos. They first extract a set of proposals of the desired object in each frame and expand the set by predictions based on the predicted object movements using optical flow. They then create a Directed Acyclic Graph (DAG) using the found proposals in all video frames. By connecting each proposal to all other proposals in the next 3 frames and finding the longest path in the constructed graph using dynamic programming, they can find the primary object in the video. Using this object alongside their proposed 3D graph segmentation, they then segment the primary object in each video frame. It is good to note that since they use the longest path method for their work, all video frames related to the object to-be-segmented should be present (which also indicates that segmenting multiple objects in one video sequence is not feasible).

We have proposed a method for automatic human subject segmentation in [3] and [6], using a coarse-to-fine shape model refinement and Grab-cut framework. It first generates some shape silhouettes through an Statistical Shape Model (SSM) [20] generator, trained with a set of hand segmented human contours. After that, by selecting one of the generated silhouettes and performing segmentation with Grab-cut and comparing the segmentation result with the generated shapes and finding the best match, the segmentation is refined until converging to the best local segmentation. Based on the parameters of the model resulting in this segmentation, the system then tries to refine the shape model and repeats this process to achieve the best segmentation result.

In contrast to the methods introduced above, which just use color feature for segmentation, we propose a method that makes use of a texture feature which makes it possible to cope with illumination changes and texture variations in different parts of the human body. We also use a human probability map to make an automated system with prior knowledge about the object-of-interest alongside the color appearance model used in super-pixel generation and image segmentation stages. Also, unlike the work by Zhang et al. [4] which needs the whole video of the object to segment

it and as a result is unfit for applications like on-line pedestrian segmentation, the proposed method performs the segmentation with just information in one frame thus achieves wider range of applications. Again, unlike our previously proposed methods in [3] and [6], multiple segmentations for model refinement has become unnecessary thanks to the usage of human probability map and super-pixels generation, thus achieving the segmentation in one iteration.

3. Proposed Method

3.1 Main Idea

Before we explain our proposed method, it would be better to first provide a brief explanation about the idea behind the proposed system.

Usually, images contain different kinds of objects either as scenery (background) or the ones that are of interest (foreground). Simple objects are usually specified by a combination of its color and texture. Complex objects like humans are hard to be defined in this concept because they have different color/texture in each part. As a result, if we map each region in the image by its color or texture, the result would become like a puzzle with multiple pieces. So it is possible to think of this as a puzzle-game problem, in which the input image would become a puzzle with multiple pieces. Our task will then become the search for correct pieces to select and find the human subject inside different pieces we have at hand. As for the way to make an image into a puzzle, using methods that turn the image into some super-pixels would be the best choice. Since we want to segment the human body, if we can create our puzzle so that the pieces related to human body are easier to pick, it would make the task much easier. There are different methods for turning the image into super-pixels like Watershed [21], SLIC Super-pixel [22], Liu et al.'s method [23], or others. Here, we have decided to use the Watershed algorithm [21] which is well-known and fast.

As it is mentioned before, human subjects usually have different colors/textures (because of their clothing and change in the color/texture because of the shadows, folding of different parts of the clothing, etc.). As a result, we considered that if we first make a puzzle from the input image in which each piece contains regions with the same texture and then try to find the human subject inside the pieces, it will be more convenient. To shape up the super-pixels to the best shape as possible, we provided the Watershed algorithm with the best region candidates based on the texture of the regions. For this, we used a texture feature map based on the work by Zhou et al. [24]

Now that we have the puzzle pieces, we have to select the right ones as human body. So, we use a human probability map which shows the probable parts that the human body might exist in the image frame. For this work, we first segmented 180 images manually and turned the segmentation result into a binary image. After that, binary images were scaled to match in either width or height, while keeping their original aspect ratio. At last, resized silhouettes were added together and normalized to create a body probability map which will be input to the system. As mentioned before, this stage is an off-line procedure which means the creation of this probability map is done before we even start the segmentation procedure. In some cases, one piece of the puzzle might contain a part of the object alongside the background. In this case, it will be possible to break them into smaller pieces and select the correct parts using this probability map. Selecting the right pieces will give us the rough shape of the human subject. This rough shape can be improved to a more accurate result by feeding the rough shape as a prior to the Grab-cut framework later in the process. It is also good to note that by using this idea, it is possible to perform the segmentation in both automatic and interactive manners.

3.2 Overveiw

The basic flow of the proposed method can be seen in Fig. 3. The inputs to the system are the image to be segmented and a human body probability map. This map represents the probability of human body parts in each part of the image. To create the map, we can manually segment some images or hand segmented contour of a set of human subjects and use the method of Cootes et al. [20] to generate some new shapes like the trained Statistical Shape Model (SSM) generator which we have previously used in [6], and combine the results into one map.

The process after data input, can be briefly explained in the following steps. More details will be presented in the following parts.

- **Super-pixel generation**: After the images have been input to the system, the first step is to convert them to some super-pixels for further processing. We use the Watershed algorithm [21] for this purpose in our work.
- **Super-pixel selection using human probability map**: After super-pixels are generated, we have to find the ones related to our object-of-interest (here, the human body) using the probability map generated before we start the segmentation.
- **Selection refinement**: Since there might be some parts of the subject in the super-pixels that were not selected, the ones that contain a part of the object are broken into smaller ones and are checked again.
- **Result refinement**: The final selected super-pixels generate the main segmentation and this segmentation is refined using Grab-cut.

3.3 Super-Pixel Generation

After an image is input to the system, the first step is to turn it into some super-pixels for further processing. In this work, we have used Watershed algorithm [21], since it has simple yet effective formulation for image segmentation which makes it ideal for our preprocessing stage. In addition, we will be able to use it multiple times if needed. When provided with some seeds, it segments the image to some

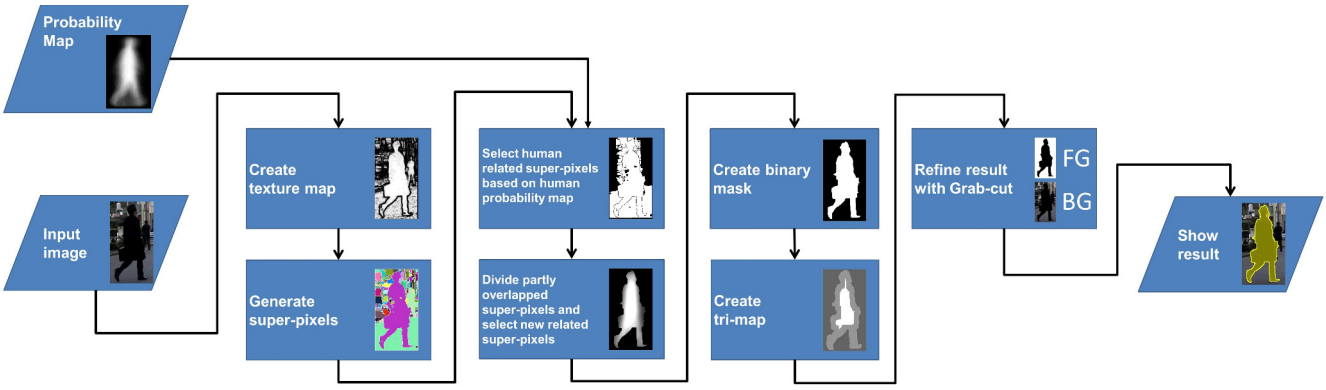


Fig. 3 Flowchart of the proposed segmentation method.

coherent regions based on the input. As some seeds are needed to define sources for segmentation, we try to provide them to the system by calculating texture features which makes the segmentation a combination of color and texture features.

For this, here, we use the same algorithm used by Zhou et al. [24] and Houhou et al. [25] which is based on the Beltrami representation [26]. The color image representation will be a 5D Riemannian manifold like

$$X(x, y) \rightarrow (X_1 = x, X_2 = y, X_3 = R(x, y), \\ X_4 = G(x, y), X_5 = B(x, y)), \quad (1)$$

where x and y are coordinates and $R, G, B(x, y)$ are color values at that coordinate. As it is mentioned by Zhou et al. [24] and Houhou et al. [25], textures are semi-local in nature so it is possible to use this property in our favor and create a feature map. For this, we change the local representation to a semi-local one using a window of size $n \times n$ [pixels] around the specific location as

$$P_R(x, y) = \left\{R(x+w_x, y+w_y) : w_x, w_y \in \left[-\frac{n-1}{2}, \frac{n-1}{2}\right]\right\}. \quad (2)$$

The same can be done for $P_G(x, y)$ and $P_B(x, y)$ which are windows in Green and Blue channels of the image, respectively. The new Beltrami representation will then become as

$$X(x, y) \rightarrow (X_1 = x, X_2 = y, X_3 = P_R(x, y), \\ X_4 = P_G(x, y), X_5 = P_B(x, y)). \quad (3)$$

Calculating the metric tensor g_{xy} for this manifold, we will have

$$g_{xy} = \begin{pmatrix} 1 + \sum_{c\in\mathbb{C}}(\partial_x P_c(x, y))^2 & \sum_{c\in\mathbb{C}} \partial_x P_c(x, y)\partial_y P_c(x, y) \\ \sum_{c\in\mathbb{C}} \partial_x P_c(x, y)\partial_y P_c(x, y) & 1 + \sum_{c\in\mathbb{C}}(\partial_y P_c(x, y))^2 \end{pmatrix}. \quad (4)$$

In above equation $\mathbb{C} = \{R, G, B\}$. Using this, we can calculate the texture feature as

(a) Subject with a single textured area

(b) Subject with multiple textured areas

Fig. 4 Example of texture feature for an input image.

$$T = \exp\left(-\frac{\det(g_{xy})}{\sigma^2}\right). \quad (5)$$

Here, the Gaussian kernel acts as a low-pass filter which gives us the ability to control the degree of details in the calculated feature by changing the value of the scaling parameter $\sigma > 0$.

Result of calculating this feature for an image is presented in Fig. 4. In Fig. 4(a), a subject is presented with an almost uniform color and texture in which, when the texture feature is calculated, the whole body will become as one block (single texture), while in Fig. 4(b), the subject has multiple textured regions as presented in the calculated texture map. This texture feature is used to give us a map of the parts of the image which are similar in the texture. As it is depicted in Fig. 4, this can be a very good asset in some cases while it can be useful in other cases.

We then convert this texture map into a binary seed map by thresholding it. Since using one constant global threshold value would sometimes connect some parts of image with different textures to each other, due to not very distinct region boundaries, we tried a local thresholding scheme to have as much details as we can. We also removed the image edges from the texture map to make sure that different regions are separated from each other to the best extent as possible. For this, we tried to calculate the threshold value in an $n \times n$ [pixels] ($n = 6$ in our work) window by calculating a weighted average inside that window.

3.4 Super-Pixels Selection Using Human Probability Map

Now that we have all image pixels as super-pixels/blocks, we have to select the ones that are related to our object of

interest (the human body in this work). For this, we use the human probability map which shows where in the image frame are most probably human body parts. Still, even with the best type of probability map, there is no guaranty that the probability map would match our current image (actually it will most probably not match exactly) so we have to set a criterion on how we select super-pixel/blocks based on the probability map. For this, we have prepared two stages.

Assume that we have m super-pixels in the image $\mathbb{S} = \{S_1, \ldots, S_m\}$. In the first stage, the super-pixel/blocks that have probability of more than the threshold P_{h_1} would be added to the set that can make the foreground mask as

$$\mathbb{M}_{\mathrm{PF}} = \{S_i|^{\exists}(x, y) \in S_i; P(x, y) \geqslant P_{\mathrm{h}_1}\}. \quad (6)$$

From these, the ones that overlap with the human probability body boundary more than P_{sp_1} percent in the probability map would be selected as the main part of a human body and put in $\mathbb{M}_{F_1}$ for creating the foreground mask. The rest of the super-pixels will be put in an auxiliary set $\mathbb{M}'_{\mathrm{PF}}$ for further process as

$$\mathbb{M}_{F_1} = \left\{S_i|\mathrm{score}(S_i) \geqslant P_{\mathrm{sp}_1}; S_i \subset \mathbb{M}_{\mathrm{PF}}\right\}, \quad (7)$$

$$\mathbb{M}'_{\mathrm{PF}} = \left\{S_i|\mathrm{score}(S_i) < P_{\mathrm{sp}_1}; S_i \subset \mathbb{M}_{\mathrm{PF}}\right\}, \quad (8)$$

where $\mathrm{score}(S_i)$ is

$$\mathrm{score}(S_i) = \frac{\#\{(x, y)|(x, y) \in S_i; P(x, y) \geqslant P_{h_1}\}}{\#\{S_i\}}. \quad (9)$$

Here, $\#\{\bullet\}$ show the number of pixels included in the set. Other super-pixels with lower probability or the ones that contain human body parts would be collected in $\mathbb{M}'_{\mathrm{PF}}$ for further processing in the second stage.

3.5 Selection Refinement

In the second stage, the super-pixels in $\mathbb{M}'_{\mathrm{PF}} = \{S'_1, \ldots, S'_q\}$ are split into smaller parts. This time, the texture feature is extracted locally on these parts to achieve more details and feed the result to Watershed. Then, the new pieces are checked against the probability map to see if there are some parts with probability more than P_{h_2} as

$$\mathbb{M}_{F_A} = \{S'_i|^{\exists}(x, y) \in S'_i; P(x, y) \geqslant P_{\mathrm{h}_2}\}. \quad (10)$$

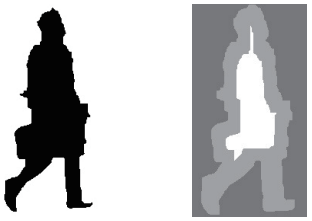
From these blocks, the ones that overlap with the human probability body boundary more than P_{sp_2} in the probability map would be selected as the main part of a human body as

$$\mathbb{M}_{F_2} = \left\{S'_i|\mathrm{score}(S'_i) \geqslant P_{\mathrm{sp}_2}; S'_i \subset \mathbb{M}_{F_A}\right\}. \quad (11)$$

The foreground mask is then created by combining the super-pixels selected in first stage and refined ones from the second stage as

$$\mathbb{M}_F = \mathbb{M}_{F_1} \cup \mathbb{M}_{F_2}. \quad (12)$$

This set is the final result for the rough shape of the human subject which will be further processed in the next stage.



(a) (b)

Fig. 5 Example of a tri-map. (a) Original mask, (b) Generated tri-map (light gray: Foreground, gray: Probably foreground, dark gray: Probably background).

3.6 Result Refinement

After selecting related super-pixels to the best possible degree, we would perform a pixel-wise refinement by Grab-cut.

In this work, we will use the Grab-cut in the same way as in our previous work [6]. We are using Grab-cut to make sure that at least some of the related parts that have not been selected in the super-pixel selection stage can be segmented and presented as the final segmentation.

For using Grab-cut at this stage, we first use the mask created by combining the silhouettes of selected super-pixels which are now stored in $\mathbb{M}_F$ into one binary image which is our base input mask for the Grab-cut segmentation stage. After that, we create a tri-map like in our previous works [3], [6], based on our base mask mentioned above. An example of a tri-map is presented in Fig. 5. After using the Grab-cut, we will have a more refined segmentation result which also includes some of the parts that have not been selected in the super-pixel selection stage.

4. Experiments

4.1 Data-Sets

Three different data-sets were prepared to test the proposed method and comparative methods.

The first data-set is a private one made from the data available in our laboratory. It consists of 180 human subjects with different sizes. The images in the dataset were extracted from high definition recorded video ($1,280 \times 720$ pixels) taken by a camera installed behind the windshield of a vehicle. It contains footage of pedestrians crossing the street or walking in pathways. Videos were taken in bright, normal, and dark places.

The second data-set has been created from Caltech Pedestrian Detection Data-set [27], [28] which is a famous data-set as a benchmark for the pedestrian detection methods. The video recording setup is the same as in our dataset. Instead, the video size is VGA (640×480 pixels) which implies that the pedestrian sizes are smaller. Since the quality of the images and pedestrian sizes are not very good in this data-set, 100 human subjects with a height more than 50 pixels were selected.

Table 1 Selection parameters and their values in the proposed method.

Parameter	Value
P_{sp_1}	0.5
P_{sp_2}	0.5
P_{h_1}	0.4
P_{h_2}	0.7

The third data-set is a subset created from the PennFudan data-set [29] which is a data-set created by groups from both Pennsylvania and Fudan Universities. This set consists of 230 human subjects with different sizes. The images were taken with stationary cameras in different places. Usually the background in the images is complex and in some cases the color similarity between foreground and background is high.

All images in the dataset were taken in day time, are RGB color images, and contain one single subject. Since in this work we want to examine how well the proposed method is capable of segmenting the boundary of a human subject, we have left the occlusion problem for our future work and focused on single subject per frame in this paper.

As for the ground-truth for evaluating the segmentation accuracy, for the first and second data-sets, ground-truth has been created by manually segmenting the human subjects and turning the results into a binary image. In case of the PennFudan data-set, the ground truth is provided for each human subject by its distributor.

4.2 Experiments Setup

As it has been mentioned before, for finding human body parts, we use a probability map. This map can provide us an estimation of the existence of human body parts in different places of a selected window. This probability map is generated off-line.

As for the thresholds to select related super-pixels to the human body parts based on the probability map in 3.5, we have experimentally set them as in Table 1.

4.3 Comparative Methods

To test the validity of the proposed method, we have compared the proposed method with some automatic and interactive segmentation methods. As for the automatic methods, first Grab-cut [13] which is famous due to simple interaction and iterative energy minimization was prepared. It has been made automatic by providing the system with the bounding box given to it. The second method was SSFSeg [6] which uses a human shape model alongside Grab-cut for automatic segmentation.

As for the interactive segmentation methods for comparison, the first one was Watershed algorithm [21] which can be considered a traditional method. It has fast response time and tends to give coherent segmentation regions. The others were Efficient Graph-cut segmentation [18], Planarcut [16], Onecut [11] and Convexity shape prior [12]. The main reason for this selection is that aside from the Watershed algorithm [21] and Convexity shape prior [12], other methods have tried a different aspect for solving the graph-based segmentation problem. Each method was briefly explained in Sect. 2. The two latter methods are new methods which have showed accurate segmentation results.

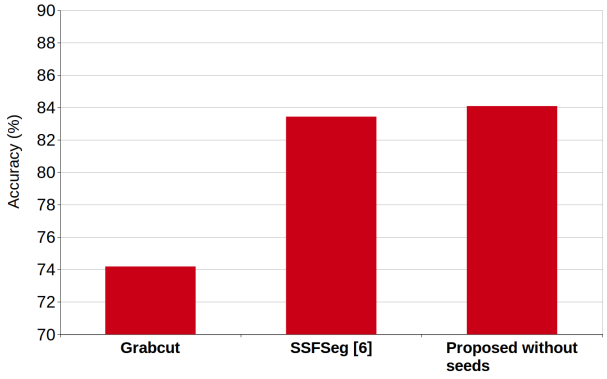


Fig. 6 Automaric segmentation: Comparison of average segmentation accuracy between different segmentation methods and the proposed method which uses human probability map and super-pixels.

4.4 Results

Some experiments have been performed to validate the proposed system. The results have been compared to the methods mentioned in 4.3. For comparison, the accuracy was calculated based on the following formula:

$$\text{Accuracy } [\%] = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \times 100. \tag{13}$$

Here, in the above equation, 'TP' represents True Positive which is the number of pixels that are correctly selected as foreground in an image, 'TN' represents True Negative which is the number of pixels that are correctly selected as background in an image, 'FP' represents False Positive which is the number of pixels that are wrongly selected as foreground in an image, and 'FN' represents False Negative which is the number of pixels that are selected as background in an image by mistake.

4.4.1 Automatic Segmentation

If the proposed method is used for automatic segmentation, the results of the system which uses human probability map and super-pixels become as presented in Fig. 6. Comparison is done between the proposed method, the automated Grab-cut [13], and SSFSeg [6]. As in the graph of Fig. 6, by combining the human probability map and super-pixels, the accuracy becomes significantly higher than the Grab-cut while it wins against the SSFSeg method.

4.4.2 Interactive Segmentation

If the proposed method is used for interactive segmentation,

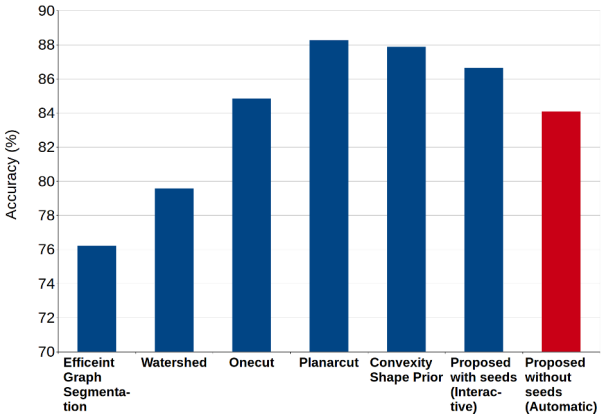
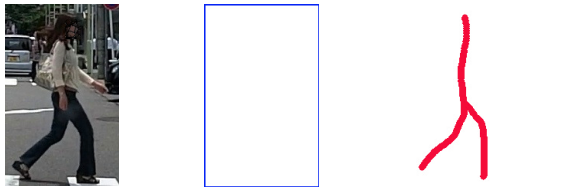


Fig. 7 Interactive segmentation: Comparison of average segmentation accuracy between different segmentation methods and the proposed method which uses human probability map and super-pixels.



(a) Input image (b) Background seeds (c) Foreground seeds

Fig. 8 Manual seed selection.

results of the system which uses human probability map and super-pixels become as presented in Fig. 7. In Fig. 7, the segmentation accuracy of the system is compared with some comparative interactive methods. It can be seen that although our method uses a relatively more simple way of doing things, the segmentation results are comparable with the state-of-the-art interactive segmentation methods. Please note that for interactive segmentation, some foreground and background seeds are necessary. For this, all of the methods were provided with the same type of seeds. As presented in Fig. 8, for background seeds, a rectangle around the picture was provided, while for the foreground, some seeds were selected manually so that they cover the basic skeletal shape of human body and almost cover the subject, for fair judgment.

5. Discussion

In this work, we proposed a system that can automatically segment human subject from an image. By converting the image into a puzzle, using a relatively simple texture feature and human probability map, we showed that it is possible to achieve good segmentation results. Using the mentioned method not only solves the main problem of Graph-cut based methods which use just a color distance feature for distinction between two regions, but also allows us segmenting the human subject automatically with more accuracy. Compared to the original Grab-cut, the accuracy improvement is significant while it is increased even compared to our previous method [6] as depicted in Fig. 6.

(a) (b) (c) (d) (e) (f) (g)

Fig. 9 Good segmentation examples from Caltech data-set: (a) Proposed method, (b) Planarcut [16], (c) Onecut [11], (d) Convexity shape prior [12], (e) Grab-cut [13], (f) Efficient Graph-cut segmentation [18], and (g) Watershed [21].

Table 2 Performance comparison between different methods in interactive mode.

Method	Accuracy (%)		
	Private	Caltech	PennFudan
Efficeint graph segmentation [18]	77.07	73.29	78.27
Watershed [21]	79.23	78.71	80.77
Onecut [11]	83.86	84.92	85.74
Planarcut [16]	89.72	87.65	87.42
Convexity shape prior [12]	88.90	88.12	86.58
Proposed with seeds	89.31	86.12	84.48

Although the proposed method is automatic by nature, it can also be used as an interactive segmentation method. The result of using the system in interactive mode compared to the same type of comparative methods is presented in Fig. 7. As it can be seen, even if the final refinement stage uses the Grab-cut method, the accuracy of the system in interactive mode is almost on par with recently proposed interactive methods, while in automatic mode, the result becomes comparable with state-of-the-art methods and much better than the traditional ones. If instead of Grab-cut, another method is used for refinement, the accuracy might improve further. Some examples of segmentation quality in comparison with other methods are depicted in Figs. 9 and 10. Comparison of all methods is also presented in Tables 2 and 3.

Still, since here we are using a simple probability map and texture feature, in some cases, the desired segmentation result is not achieved. An example of this is presented in Fig. 11. The main reason of failure for Fig. 11(a) is the wrong probability prediction by the human probability map. The reason for Fig. 11(b) is the miscalculation in the texture because of the similarity between the color and texture of the foreground object and a part of the background which leads to creation of a super-block. When this super-block

(a) (b) (c) (d) (e) (f) (g)

Fig. 10 Good segmentation from PennFudan data-set: (a) Proposed method, (b) Planarcut [16], (c) Onecut [11], (d) Convexity shape prior [12], (e) Grab-cut [13], (f) Efficient Graph-cut segmentation [18], and (g) Watershed [21].

Table 3 Performance comparison in accuracy (%) between different methods in automatic mode.

Method	Dataset		
	Private	Caltech	PennFudan
Proposed without seeds	87.36	85.64	79.24
SSFSeg [6]	83.66	86.03	80.60
Grabcut [13]	72.08	71.45	79.03

is checked against P_{sp_1}, it will be considered as part of the background.

We mentioned before how we create our probability map in 3.1. It is also good to mention that the off-line procedure of creating the map can be accompanied with an on-line updating scheme to improve the map by adding the mask of segmented subjects. Although this is a possible way for improvement, in this work, we just incorporated the off-line stage. Also it is good to note that even though different methods can be used to turn an image into super-pixels, the Watershed algorithm provided satisfactory results for our work based on the given texture seeds. As depicted in Fig. 12, by just looking at the super-pixels, the boundary of human body is recognizable. Still, using other methods might improve the system results and we intend to do this as one of the future works.

Although the proposed method in this work performs the segmentation of the image in one iteration in contrast to our previous methods [3], [6], the segmentation result is almost the same on Caltech and PennFudan datasets while

(a) Failure due to probability map (b) Failure due to texture map

Fig. 11 Two examples of segmentation failure from Caltech data-set.

Fig. 12 Watershed texture based super-pixels generation results.

improving by 4% on the private dataset. The main reason for improvement is the use of probability map and super-pixels instead of initializing a shape model on the image that might not match completely. Also the computation time has been significantly reduced. For example, the proposed method can segment all 180 images of the private dataset in 3.08 minutes (1.03 seconds per image) while the Grab-cut performs the task in 39.6 seconds (0.22 seconds per image) and SSFSeg [6] does it in 3.35 hours (67 seconds per image).

6. Conclusion

In this paper, we have proposed an automatic method for human subject segmentation in single shot frames with good accuracy comparable to some state-of-the-art methods. The main idea is to segment an image into multiple super-pixels and then try to find super-pixels which are related to human body parts using a human body probability map. Although the used probability map is relatively simple, the system shows promising results in segmentation. Also, even though the system is automatic, we have confirmed the possibility of using it interactively.

Still, there are some problems to be considered for further work. The first is to use other types of probability maps which can provide more accurate information for human body. The second is to find a better refinement algorithm instead of the Grab-cut. The refinement stages and the Grab-cut refinement in complex images sometimes become insignificant which implies that using the methods which do

not solely rely on color features might be more useful. It is good to note that, even with these problems, the proposed method shows a significant improvement compared to the original Grab-cut algorithm and is also automatic. As for the future work, we would like to:

- Find a better method to generate and use the probability map.
- Unify multiple stages in one framework.
- Apply the proposed method to other frameworks or methods like Planarcut [16].
- Perform some code optimization.

Acknowledgments

Parts of this research were supported by MEXT, Grant-in-Aid for Scientific Research.

References

[1] Y. Li, J. Sun, and H.-Y. Shum, "Video object cut and paste," ACM Trans. Graphic., vol.24, no.3, pp.595–600, July 2005.
[2] E.N. Mortensen and W.A. Barrett, "Intelligent scissors for image composition," Proc. 22nd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH'95, pp.191–198, Aug. 1995.
[3] E. Pourjam, I. Ide, D. Deguchi, and H. Murase, "Segmentation of human instances using grab-cut and active shape model feedback," Proc. 13th IAPR Int. Conf. Machine Vision Applications (MVA), pp.77–80, May 2013.
[4] D. Zhang, O. Javed, and M. Shah, "Video object segmentation through spatially accurate and temporally dense extraction of primary object regions," Proc. 2013 IEEE Conference on Computer Vision and Pattern Recognition, pp.628–635, June 2013.
[5] V. Gulshan, V. Lempitsky, and A. Zisserman, "Humanising GrabCut: Learning to segment humans using the Kinect," Proc. 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pp.1127–1133, June 2011.
[6] E. Pourjam, D. Deguchi, I. Ide, and H. Murase, "Statistical shape feedback for human subject segmentation," IEEJ Trans. EIS, vol.135, no.8, pp.1000–1008, Aug. 2015.
[7] S. Prakash, S. Das, and R. Abhilash, "Snakecut: An integrated approach based on active contour and grabcut for automatic foreground object segmentation," Electronic Letters on Comput. Vision and Image Anal. (ELCVIA), vol.6, no.3, pp.13–28, 2007.
[8] J. Shi and J. Malik, "Normalized cuts and image segmentation," IEEE Trans. Pattern Anal. Mach. Intell., vol.22, no.8, pp.888–905, Aug. 2000.
[9] Y. Li, J. Sun, C.-K. Tang, and H.-Y. Shum, "Lazy snapping," ACM Trans. Graphic., vol.23, no.3, pp.303–308, Aug. 2004.
[10] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," Int. J. Comput. Vision, vol.1, no.4, pp.321–331, Jan. 1988.
[11] M. Tang, L. Gorelick, O. Veksler, and Y. Boykov, "GrabCut in one cut," Proc. 2013 IEEE International Conference on Comput. Vision, pp.1769–1776, Dec. 2013.
[12] L. Gorelick, O. Veksler, Y. Boykov, and C. Nieuwenhuis, "Convexity shape prior for segmentation," Proc. Comput. Vision, ECCV 2014, vol.8693, pp.675–690, 2014.
[13] C. Rother, V. Kolmogorov, and A. Blake, ""GrabCut": Interactive foreground extraction using iterated graph cuts," ACM Trans. Graphic., vol.23, no.3, pp.309–314, Aug. 2004.
[14] Y.Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images," Proc. Eighth IEEE International Conference on Comput. Vision, ICCV 2001, pp.105–112, July 2001.
[15] D. Weinland, R. Ronfard, and E. Boyer, "A survey of vision-based methods for action representation, segmentation and recognition," Comput. Vis. Image Und., vol.115, no.2, pp.224–241, Feb. 2011.
[16] F.R. Schmidt, E. Toppe, and D. Cremers, "Efficient planar graph cuts with applications in computer vision," Proc. 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp.351–356, 2009.
[17] K. Weihe, "Edge-disjoint (s,t)-paths in undirected planar graphs in linear time," J. Algorithm., vol.23, no.1, pp.121–138, April 1997.
[18] P.F. Felzenszwalb and D.P. Huttenlocher, "Efficient graph-based image segmentation," Int. J. Comput. Vision, vol.59, no.2, pp.167–181, Sept. 2004.
[19] P.F. Felzenszwalb, R.B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," IEEE Trans. Pattern Anal. Mach. Intell., vol.32, no.9, pp.1627–1645, Sept. 2010.
[20] T.F. Cootes, C.J. Taylor, D.H. Cooper, and J. Graham, "Active shape models-their training and application," Comput. Vis. Image Und., vol.61, no.1, pp.38–59, Jan. 1995.
[21] F. Meyer and S. Beucher, "Morphological segmentation," J. Vis. Commun. Image R., vol.1, no.1, pp.21–46, Sept. 1990.
[22] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," IEEE Trans. Pattern Anal. Mach. Intell., vol.34, no.11, pp.2274–2282, Nov. 2012.
[23] M.-Y. Liu, O. Tuzel, S. Ramalingam, and R. Chellappa, "Entropy rate superpixel segmentation," Proc. 2011 IEEE International Conference on Computer Vision Library 2011, pp.2097–2104, 2011.
[24] H. Zhou, J. Zheng, and L. Wei, "Texture aware image segmentation using graph cuts and active contours," Pattern Recogn., vol.46, no.6, pp.1719–1733, June 2013.
[25] N. Houhou, J.P. Thiran, and X. Bresson, "Fast texture segmentation based on semi-local region descriptor and active contour," Numerical Mathematics: Theory, Methods and Applications, vol.2, no.4, pp.445–468, Sept. 2009.
[26] N. Sochen, R. Kimmel, and R. Malladi, "A general framework for low level vision," IEEE Trans. Image Process., vol.7, no.3, pp.310–318, March 1998.
[27] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: A benchmark," Proc. 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp.304–311, Nov. 2009.
[28] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," IEEE Trans. Pattern Anal. Mach. Intell., vol.34, no.4, pp.743–761, April 2012.
[29] L. Wang, J. Shi, G. Song, and I.F. Shen, "Object detection combining recognition and segmentation," Proc. 2009 Asian Conf. Comput. Vision (ACCV), pp.189–199, Nov. 2007.

Esmaeil Pourjam received his BEng in Electronics form Shahid Beheshti University of Tehran, Iran in 2009 and MEng in Mechatronics from Semnan University, Iran in 2011 and is currently pursuing his PhD in Information Science as a MEXT scholarship student in Graduate School of Information Science of Nagoya University, Japan. His main interests are robotics, computer vision and intelligent systems. His research is currently focused on human and pedestrian segmentation methods with applications in intelligent vehicles and human-machine interfaces.

Daisuke Deguchi received his BEng and MEng degrees in Engineering and a PhD degree in Information Science from Nagoya University, Japan, in 2001, 2003, and 2006, respectively. He is currently an Associate Professor in Information Strategy Office, Nagoya University, Japan. He is working on the object detection, segmentation, recognition from videos, and their applications to ITS technologies, such as detection and recognition of traffic signs.

Ichiro Ide received his BEng, MEng, and PhD from The University of Tokyo in 1994, 1996, and 2000, respectively. He became an Assistant Professor at the National Institute of Informatics, Japan in 2000. Since 2004, he has been an Associate Professor at Nagoya University. He had also been a Visiting Associate Professor at National Institute of Informatics from 2004 to 2010, an Invited Professor at Institut de Recherche en Informartique et Systèmes Aléatoires (IRISA), France in 2005, 2006, and 2007, a Senior Visiting Researcher at ISLA, Instituut voor Informatica, Universiteit van Amsterdam from 2010 to 2011. His research interest ranges from the analysis and indexing to retargeting of multimedia contents, especially in large-scale broadcast video archives, mostly on news, cooking, and sports contents. He has been serving on program committees at conferences such as ACMMM, CVPR, and ICCV. He is a senior member of IPS Japan, and member of JSAI, IEEE, and ACM.

Hiroshi Murase received the BEng, MEng, and PhD degrees in Electrical Engineering from Nagoya University, Japan. In 1980 he joined the Nippon Telegraph and Telephone Corporation (NTT). From 1992 to 1993 he was a visiting research scientist at Columbia University, New York. From 2003 he is a Professor of Nagoya University, Japan. He was awarded the IEICE Shinohara Award in 1986, the Telecom System Award in 1992, the IEEE CVPR (Conference on Computer Vision and Pattern Recognition) Best Paper Award in 1994, the IPS Japan Yamashita Award in 1995, the IEEE ICRA (International Conference on Robotics and Automation) Best Video Award in 1996, the Takayanagi Memorial Award in 2001, the IEICE Achievement Award in 2002, and the Ministry Award from the Ministry of Education, Culture, Sports, Science and Technology in 2003. Dr. Murase is IEEE Fellow and a member of the IPS Japan.