Computational Visual Media
https://doi.org/10.1007/s41095-017-0098-0 Vol. 4, No. 1, March 2018, 43–54

Research Article

# Transferring pose and augmenting background for deep human-image parsing and its applications

**Takazumi Kikuchi**[1], **Yuki Endo**[1](✉), **Yoshihiro Kanamori**[1](✉), **Taisuke Hashimoto**[1], **and Jun Mitani**[1]



**Abstract** Parsing of human images is a fundamental task for determining semantic parts such as the face, arms, and legs, as well as a hat or a dress. Recent deep-learning-based methods have achieved significant improvements, but collecting training datasets with pixel-wise annotations is labor-intensive. In this paper, we propose two solutions to cope with limited datasets. Firstly, to handle various poses, we incorporate a pose estimation network into an end-to-end human-image parsing network, in order to transfer common features across the domains. The pose estimation network can be trained using rich datasets and can feed valuable features to the human-image parsing network. Secondly, to handle complicated backgrounds, we increase the variation in image backgrounds automatically by replacing the original backgrounds of human images with others obtained from large-scale scenery image datasets. Individually, each solution is versatile and beneficial to human-image parsing, while their combination yields further improvement. We demonstrate the effectiveness of our approach through comparisons and various applications such as garment recoloring, garment texture transfer, and visualization for fashion analysis.

**Keywords** image segmentation; semantic segmentation; human-image parsing; deep convolutional neural network

## 1 Introduction

Human-image parsing is the image-processing task of assigning semantic labels to human body parts and clothing regions including the face, arms, and legs, or a hat, dress, etc. This task plays a crucial role in various applications in computer graphics and computer vision, e.g., virtual fitting systems [1], clothing retrieval [2], and recommendation [3, 4].

Recent human-image parsing methods using deep learning have exhibited significant improvements. Such methods require a sufficiently large training dataset in order to cope with various human poses and complicated background images. If sufficient training data cannot be obtained, performance is degraded. Training data are usually produced by manually annotating images with pixel-wise labels, which is quite tedious and costly even if we use crowd sourcing. This leads to the following research question: *"Can we improve human-image parsing using a limited training dataset?"*

In this paper, we answer the research question through the following two solutions. Firstly, to handle various poses, we exploit transfer learning with human pose estimation. For pose estimation, the required data are joint-wise annotations, which are easier to collect than pixel-wise annotations needed for human-image parsing. The key idea is to integrate human pose estimation into an end-to-end network model for human-image parsing, in order to transfer information from human pose estimation to the human-image parsing network, across domains that share a common feature space. While this idea can be accomplished in various ways, as a proof of concept, we use relatively-simple, state-of-the-art convolutional neural networks (CNNs)

1 University of Tsukuba, 1-1-1 Tennohdai, Tsukuba City, Ibaraki, Japan. E-mail: T. Kikuchi, kikuchi@npal.cs.tsukuba.ac.jp; Y. Endo, endo@cs.tsukuba.ac.jp (✉); Y. Kanamori, kanamori@cs.tsukuba.ac.jp (✉); T. Hashimoto, hashimoto@npal.cs.tsukuba.ac.jp; J. Mitani, mitani@cs.tsukuba.ac.jp.

Manuscript received: 2017-09-08; accepted: 2017-11-08

for human pose estimation [5] and human-image parsing [6]. Although other deep-learning-based methods for human-image parsing do not consider pose information explicitly, the explicit integration of this human-specific knowledge is beneficial to human-image parsing. Secondly, we propose a simple yet effective data augmentation method for human-image parsing. To handle various background images, we automatically replace the backgrounds of existing labeled data with new background images obtained from public large-scale datasets for scene recognition, e.g., Ref. [7]. While each technique boosts the accuracy of human-image parsing by itself, a combination of both yields further improvement. We demonstrate the effectiveness of our approach by quantitative and qualitative comparisons with existing CNN-based methods. We also show several applications such as garment recoloring, garment texture transfer, and visualization for fashion analysis using our human-image parsing results.

## 2 Related work

Early methods for human-image parsing used conditional random fields (CRFs). Yamaguchi et al.'s seminal work on human-image parsing mutually learns human pose and segmentation [8]. They later improved the performance of human-image parsing by using tagging information from similar images retrieved by $k$-nearest neighbor search [9]. Simo-Serra et al. [10] also improved on Ref. [8] by encoding the global appearance and shape of persons by considering the positions and shapes of superpixels. Instead of using CRFs, Dong et al. [11] presented a novel hybrid parsing model, which unifies human-image parsing and pose estimation. Such a unified approach has also been applied to video [12].

In recent years, deep-learning-based methods have achieved significant improvements. Liang et al. [13] first used a CNN for human-image parsing. Later, they developed a novel network called Contextualized CNN (Co-CNN), which appends the output of each layer to global image features [6]. Liu et al. [14] proposed a matching CNN, which uses a target image as input and a similar image retrieved by $k$-nearest neighbor search.

Human-image parsing is a specific semantic object segmentation task for which various CNN-based methods have been proposed [15–20]. In particular, some CNN-based methods use training datasets with different domains. Dai et al. [21] proposed use of multi-task network cascades (MNCs), which combine multiple tasks (object detection, mask extraction, and semantic labeling) in a single network. Hong et al. [22] proposed learning semantic segmentation and image classification in the same network. Papandreou et al. [23] developed an expectation-maximization method for training based on data with large amounts of weak annotation such as many bounding boxes, image level labels, and a small number of pixel-level semantic segmentation data.

Several pose estimation methods use CNNs, e.g., using a simple model consisting of convolution and pooling layers [5], by incorporating prior geometric knowledge of the body into a CNN framework [24] or by inferring correlation between joints [25].

The main contributions of this paper are to integrate human pose estimation into human-image parsing as well as to increase background image variation automatically. Both approaches can be easily integrated into existing deep-learning methods to improve human-image parsing even when only a small dataset of pixel-wise annotations is available. Although human poses have previously been exploited in CRF-based methods [8, 9] and other methods [11, 12], ours is the first attempt to explicitly integrate such information into deep neural networks, to the best of our knowledge.

## 3 Background

This section reviews existing methods for human pose estimation [5] and human-image parsing [6], components of our architecture.

### 3.1 Convolutional pose machines

Convolutional pose machines [5] define a partial network consisting of a convolutional layer and a pooling layer as one stage to obtain a heatmap for each joint. This stage is repeated multiple times to improve output human poses represented as heatmaps. For example, the pose estimation unit in Fig. 1 has three stages. The network is learned by minimizing loss functions for the multiple stages to avoid the vanishing gradient problem due to the deep architecture. This network structure can be easily

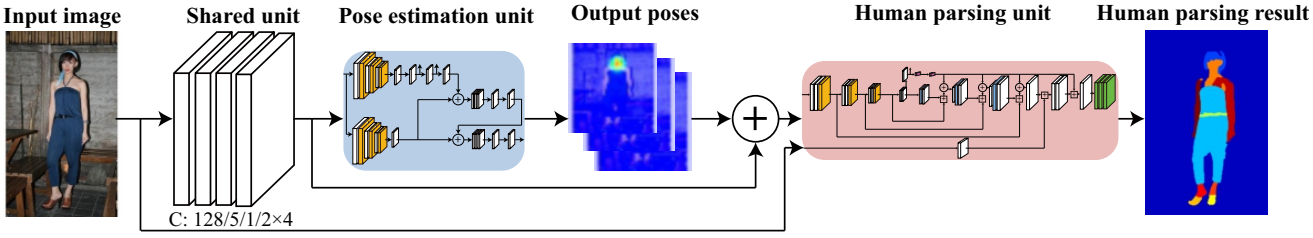



**Fig. 1** Network model. Given an input image, image features are extracted in the shared unit. Human pose is estimated as joint-wise heatmaps in the pose estimation unit. Outputs of the shared and pose estimation units are concatenated. The human-image parsing unit outputs a labeled result using the concatenated features. (More detail is given in the Electronic Supplementary Material (ESM).)

integrated into our framework because it simply consists of convolutional and pooling layers, enabling end-to-end learning.

### 3.2 Contextualized CNN

Contextualized CNN (Co-CNN) [6] is a neural network devised to improve the performance of human-image parsing. It learns global as well as local features on the basis of cross-layer context and global image-level context. As shown in the human-image parsing unit in Fig. 1, the cross-layer context is captured by using skip connections between down-convolution and up-convolution layers from fine to coarse scales. On the other hand, the global image-level context is captured by the fully connected layers, which predict image-level labels for the entire image. The predicted image-level labels are subsequently concatenated with each input to be unpooled. In addition, Co-CNN accounts for the local superpixel context. To capture this context, it has three layers at the end of the network, for within-superpixel smoothing, cross-superpixel neighbor voting, and pixel-wise prediction. These layers retain local label consistency by use of superpixels.

## 4 Proposed method

This section describes our network that transfers information of human pose estimation to the human-image parsing domain, as well as our approach for background augmentation.

### 4.1 Transferring pose estimation information

To deal with various human poses, our approach first estimates human pose before human-image parsing, and assigns pose labels to each pixel of the input image. Figure 1 shows our network model. Firstly, the input image is fed into the shared unit, and low and mid-level features are extracted. The shared unit consists of four convolutional layers with kernel size $5 \times 5$, stride 1, padding 2, and 128 output channels. Features extracted in the shared unit are fed into the pose estimation unit. The network structure of the pose estimation unit follows the network of Wei et al. [5]. In this network, a partial network consisting of a convolutional layer and a pooling layer is defined as one stage; human pose estimation is improved gradually by repeating this stage multiple times. The outputs of the pose estimation unit and shared unit are concatenated and fed into the human-image parsing unit, which finally outputs a labeled image. The human-image parsing unit uses the Co-CNN model [6], which outputs a global distribution of labels through the fully connected layers after the convolutional layers. The human-image parsing result is calculated via the deconvolutional layers, and the final result is obtained by superpixel-based smoothing. Further details of each unit are given in the ESM.

#### *4.1.1 Learning*

We train the proposed network using pose estimation and human-image parsing datasets. For the pose estimation dataset, the parameters $\theta_{\mathrm{s}}$ and $\theta_{\mathrm{p}}$ of the shared unit and pose estimation unit are optimized by minimizing the following error function:

$$E_{\mathrm{p}} = \sum_{\boldsymbol{b}_i, \boldsymbol{b}_l \in \mathcal{B}} \sum_{t=1}^{T} \sum_{j=1}^{J} \| \boldsymbol{b}_l^j - \boldsymbol{B}_t^j(\boldsymbol{b}_i; \theta_{\mathrm{s}}, \theta_{\mathrm{p}}) \|_2^2 \qquad (1)$$

where $\mathcal{B}$ is the pose estimation dataset containing each input image $\boldsymbol{b}_i$ and its ground-truth joint heatmap $\boldsymbol{b}_l$. $T$ is the number of repeating stages, $J$ is the number of joints to be estimated, and $\boldsymbol{B}$ is the joint heatmap estimated by the pose estimation unit. The ground-truth joint heatmaps are generated using a Gaussian function $\exp(-\|x - \mu_j\|^2/\sigma^2)$ of position $\boldsymbol{x}$, where $\mu_j$ is the position of joint $j$ and $\sigma = 2$.

For the human-image parsing dataset, instead of the error function Eq. (1) defined for pose estimation, the parameter $\theta$ of the entire network is optimized by minimizing the following error function:

$$\begin{cases} E_l = E_l^{\text{orig}} + E_l^{\text{accel}} \\ E_l^{\text{orig}} = -\sum\limits_{\{\boldsymbol{d}_i, \boldsymbol{d}_l\} \in \mathcal{D}} \sum\limits_{j}^{M} \sum\limits_{k}^{L} \boldsymbol{d}_{l_{jk}} \ln(\boldsymbol{F}_{jk}(\boldsymbol{d}_i; \theta)) \\ \qquad\qquad + \sum\limits_{\{\boldsymbol{d}_i, \boldsymbol{d}_{l'}\} \in \mathcal{D}} ||\boldsymbol{d}_{l'} - \boldsymbol{H}(\boldsymbol{d}_i; \theta)||^2 \\ E_l^{\text{accel}} = -\sum\limits_{\{\boldsymbol{d}_i, \boldsymbol{d}_l\} \in \mathcal{D}} \sum\limits_{j}^{N} \sum\limits_{k}^{L} \boldsymbol{d}_{l_{jk}} \ln(\boldsymbol{G}_{jk}(\boldsymbol{d}_i; \theta)) \end{cases} \tag{2}$$

where $E_l^{\text{orig}}$ is similar to the error function used in Ref. [6]. Adding $E_l^{\text{accel}}$ accelerates convergence. $\mathcal{D}$ is the human-image parsing dataset containing each input image $\boldsymbol{d}_i \in \boldsymbol{R}^{h \times w \times c}$, the corresponding ground-truth labeled image $\boldsymbol{d}_l \in \boldsymbol{R}^{h \times w \times L}$, and global class distribution $\boldsymbol{d}_{l'} \in \boldsymbol{R}^L$ for the entire image; $w$ and $h$ are the width and height of each input image, $c$ is its number of channels, $M$ is its number of superpixels, $N$ is its number of pixels, and $L$ is the number of class labels ($L = 18$, as in Ref. [6]). $\boldsymbol{F}$ is the output of the human-image parsing unit, $\boldsymbol{G}$ is the output before superpixel processing of the human-image parsing unit, and $\boldsymbol{H}$ is the output after the fully connected layers.

To train the network, we divide one epoch of the learning procedure into two steps. In the first step, we optimize the model parameters of the shared unit and pose estimation unit on the basis of $E_{\text{p}}$ by using the pose estimation dataset. In the second step, we optimize the model parameters of the "entire" network on the basis of $E_l$ by using the human-image parsing dataset. We used the Momentum SGD optimizer with a learning rate of 0.001, momentum term of 0.9, and weight decay term of 0.0005.

### 4.2 Augmenting background variations

To make human-image parsing robust to background variations, we augment the background patterns in the training dataset. Specifically, we cut out foreground human regions from labeled images and paste them over new background images obtained from a scenery image dataset.

Figure 2 illustrates how we augment the dataset. Inputs are a pair of a cut-out human image and its corresponding label map (see Fig. 2(b)), and a new

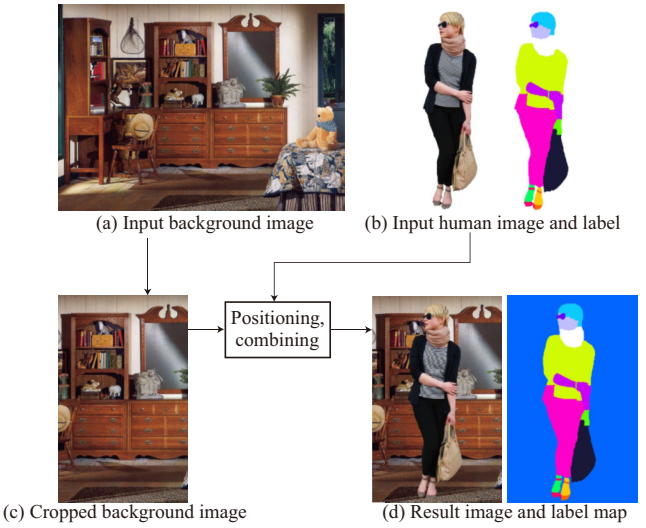



**Fig. 2** Background augmentation procedure.

background image (see Fig. 2(a)). Because most background images are wider than tall, we trim them so that their aspect ratios are consistent with the human cut-out images (see Fig. 2(c)). Figure 3 shows the procedure in detail. First, in the original dataset for human-image parsing, we calculate the mean and standard deviation of the relative width and relative position of the human region in each image. We then determine the new trimmed background width and position of the cut-out human image according to normal distributions defined by these statistics. Using the determined width, we crop the left and right sides of the input background image. The position of cropping can also be determined randomly. Finally, we paste the cut-out human image onto the cropped background while placing the human label map at the same position (Fig. 2(d)). This technique reasonably scales human images. Our data augmentation plays an important role in increasing background variation to

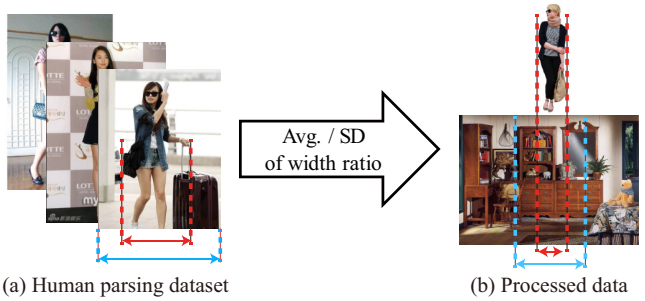



**Fig. 3** Details of background image trimming.

improve the performance of human-image parsing, as demonstrated in the evaluation section.

## 5 Evaluation

This section describes experiments to compare the proposed approach with the baseline method, Co-CNN [6].

### 5.1 Settings

The pose estimation unit (see Section 3.1) contained six stages in Wei et al.'s original method [5], but we used three in our unit, in order to reduce computational time and GPU memory footprint. For human-image parsing, Liang et al.'s method [6] uses several types of features to calculate the similarity between superpixels. However, we only use the RGB feature because the implementation details of other features, e.g., the HOG feature for each superpixel, are not clearly presented in their paper and their source code is not publicly available. We implemented our method and the baseline method in Python using the Chainer library, and ran it on a PC with an NVIDIA GeForce GTX 1080 GPU. Calculation time for the model as a whole was about 0.028 s averaged over 1000 test data.

As the human-image parsing dataset, we used the ATR dataset [13]. It contains 7702 images, of which we used 6000 images for training, 702 images for validation, and 1000 images for testing. As the background dataset used for data augmentation, we randomly selected 6000 images from an indoor scene recognition dataset [7] and doubled the 6000 training images of the ATR dataset by synthesis. Note that, although unnatural backgrounds might be selected because of random selection from the dataset, even unnatural backgrounds have a correct semantic label (i.e., "background (bg)"), and thus help to increase variation in the combination of human and background images. As the pose estimation dataset, we used the MPII Human Pose dataset [26]. It contains 24,984 images, of which we used 10,298 images annotated as training data. We included only one human in the dataset for learning.

Like Ref. [6], we used $100 \times 150$ images as input to the baseline method, and when we used only the proposed data augmentation method. When using the proposed network including the pose estimation part, we used $256 \times 256$ images as input as the size of the input image must be a power of two so that the size of the image output by pose estimation does not change. All generated results were finally resized to their original size.

### 5.2 Evaluation methods

We compared the baseline method (Co-CNN) [6], our data augmentation method (DA), and the proposed network, which uses pose estimation information (PE). As evaluation metrics, we used accuracy, precision, recall, and F1. To verify the effectiveness of the proposed method depending on the amount of training data, we conducted experiments by training with different amounts of training data for human-image parsing, 1000 and 6000 images. We stopped learning when the error function in Eq. (2) converged and used the models with maximum accuracy for validation.

Note that faithful reproduction of the Co-CNN performance [6] is almost impossible for anyone but the authors of Ref. [6]; firstly, their source code is unavailable. Secondly, the choices of test data, training data, and validation data are not revealed. Thirdly, several implementation details are missing, as mentioned in Section 5.1. Nonetheless, our goal here is to answer our research question; we demonstrate that our method designed for a small dataset outperforms the baseline.

### 5.3 Results

Table 1 shows the performance of each method for the test data. The results for data augmentation show that performance improved over those of Co-CNN when 1000 training images were used. On the other hand, the performance difference was marginal with 6000 training images. This is natural because the more training images, the more variation in background images. Recall that our purpose is to improve the performance of human-image parsing when limited training data are available, and our

**Table 1** Performance of each method using 1000 and 6000 training images

| Method | Images | Accuracy | Precision | Recall | F1 |
|---|---|---|---|---|---|
| Co-CNN | 1000 | 82.07 | 79.14 | 82.07 | 80.19 |
| DA | 1000 | 83.27 | 81.64 | 83.28 | 81.81 |
| PE | 1000 | 84.77 | 83.06 | 84.77 | 83.49 |
| DA+PE | 1000 | **85.18** | **84.67** | **85.18** | **84.43** |
| Co-CNN | 6000 | 86.15 | 84.79 | 86.15 | 84.95 |
| DA | 6000 | 86.16 | 84.78 | 86.16 | 85.15 |
| PE | 6000 | 88.31 | 88.82 | 89.00 | 88.41 |
| DA+PE | 6000 | **89.73** | **89.46** | **89.73** | **89.37** |

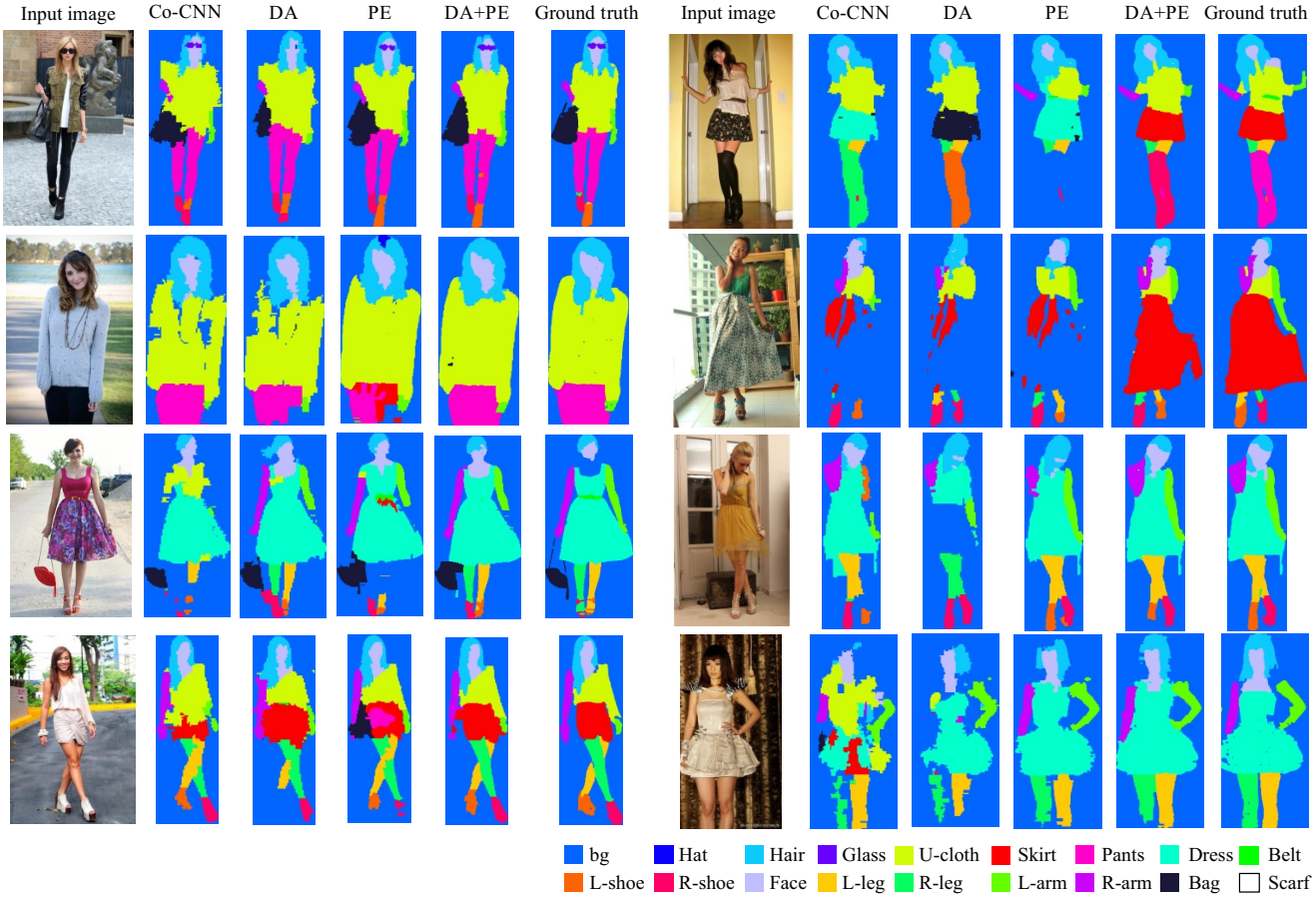



**Fig. 4** Segmentations produced by each method.

background augmentation approach is effective for this purpose.

When transferring pose estimation information to the human-image parsing part, the performance improved for both 1000 and 6000 training images. Furthermore, as shown in Table 2, a similar tendency was confirmed for F1 for each class. In particular, with few training images, our data augmentation method outperformed the baseline for multiple classes, including the background (bg) class. Even when many training images were used, the proposed network based on pose estimation significantly outperformed the baseline for all labels except scarf.

Figure 4 qualitatively compares the results for various inputs. It demonstrates that our data augmentation method successfully classified the background and foreground, and the proposed network based on pose estimation accurately extracted human body parts.

## 6 Applications

We have exploited the results of our human-image parsing method in various high-level tasks such as garment recoloring, retexturing, and visualization for fashion analysis.

### 6.1 Garment recoloring

We implemented a simple application to

**Table 2** F1 score for each class, for each method

| Method | Images | bg | Hat | Hair | Glass | U-cloth | Skirt | Pants | Dress | Belt | L-shoe | R-shoe | Face | L-leg | R-leg | L-arm | R-arm | Bag | Scarf |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Co-CNN | 1000 | 93.89 | **4.17** | 52.46 | 4.08 | 51.40 | 9.63 | 37.41 | 26.66 | 4.14 | 25.44 | 25.57 | 61.42 | 42.66 | 41.32 | 31.22 | 27.72 | 12.81 | 0.46 |
| DA | 1000 | 94.76 | 3.11 | 57.70 | **9.39** | 55.02 | 9.11 | 32.32 | **32.48** | **4.32** | 30.33 | 30.95 | 64.23 | 47.41 | 46.55 | 33.03 | 34.19 | 15.30 | **1.03** |
| PE | 1000 | 95.54 | 0.29 | 61.34 | 0.52 | 60.96 | 21.48 | 40.65 | 30.49 | 0.00 | **38.26** | 35.75 | 72.23 | 48.85 | 50.18 | **41.94** | 39.14 | 28.93 | 0.00 |
| DA+PE | 1000 | **96.18** | 0.50 | **63.06** | 0.00 | **62.88** | **36.31** | **49.50** | 16.23 | 0.46 | 36.41 | **38.86** | **73.22** | **54.51** | **54.64** | 41.65 | **43.45** | **34.54** | 0.00 |
| Co-CNN | 6000 | 95.73 | 18.15 | 66.37 | 14.04 | 64.09 | 23.83 | 49.39 | 37.26 | 7.05 | 39.77 | 40.59 | 74.08 | 58.13 | 58.12 | 48.27 | 47.39 | 35.90 | **3.56** |
| DA | 6000 | 95.93 | 0.15 | 68.28 | 8.00 | 63.89 | 28.76 | 50.83 | 36.67 | 4.50 | 35.96 | 39.70 | 73.62 | 57.82 | 57.54 | 47.50 | 47.01 | 36.99 | 0.37 |
| PE | 6000 | 97.20 | 40.30 | 74.71 | 18.87 | 69.64 | 41.57 | **61.55** | 50.75 | **21.56** | 44.85 | 45.09 | 80.54 | 65.39 | 64.31 | 62.16 | 61.70 | 48.58 | 0.03 |
| DA+PE | 6000 | **97.55** | **45.58** | **77.22** | **31.31** | **74.46** | **47.49** | 61.40 | **51.67** | 16.73 | **45.72** | **46.09** | **82.44** | **67.11** | **66.89** | **65.07** | **63.25** | **53.32** | 0.10 |

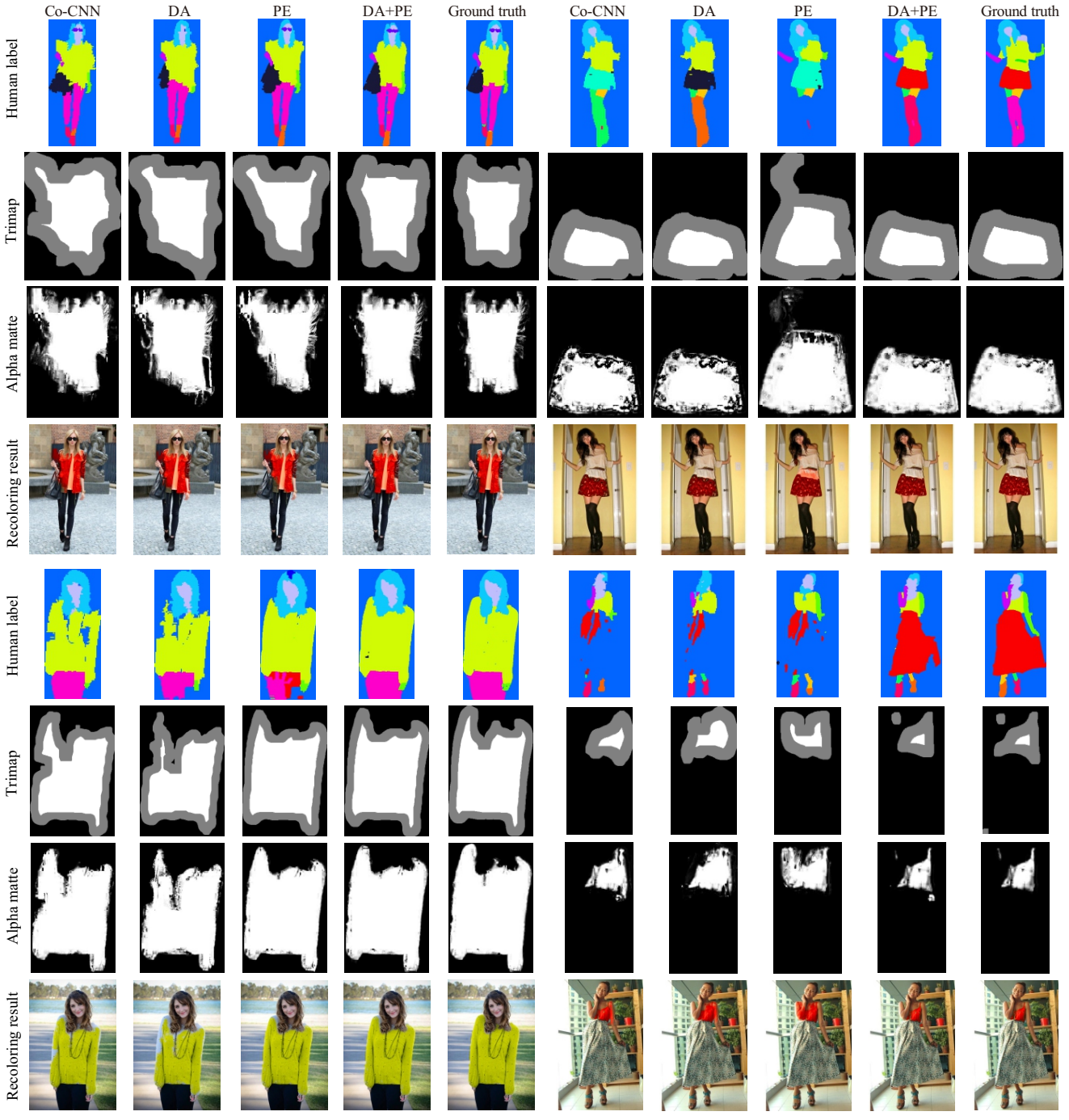
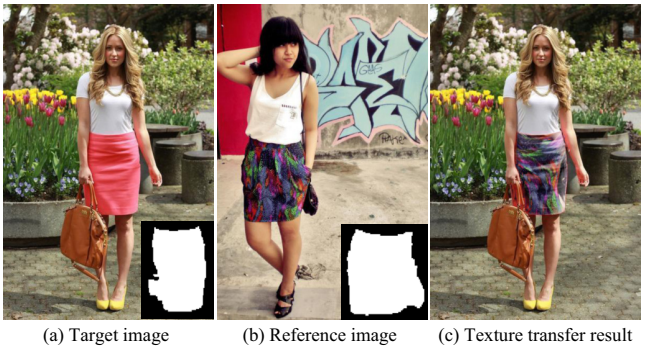



**Fig. 5** Garment recoloring. To refine each extracted garment region, we generate a trimap using morphological operations and apply alpha matting. We then change the color of the region by replacing *ab* channels in Lab color space with a user-specified color. We also apply boundary smoothing as post processing. The input images are shown in the first two rows of Fig. 4.

automatically change the colors in specific garment regions obtained by human-image parsing. To refine an extracted garment region, we first generate an alpha matte from a trimap marking definite background, definite foreground, and uncertain regions. This trimap is generated by applying morphological operators (erosion and dilation) to the specified garment region. We used the state-of-the-art method [27] for alpha matting. We then changed the color in the alpha matte region by replacing the *ab* channels in CIE Lab color space with a user-specified color. Finally, the recolored region is further refined by smoothing the colors around the matte contours with a joint bilateral



(a) Target image (b) Reference image (c) Texture transfer result

**Fig. 6** Garment texture transfer. We calculate texture coordinates both for the (a) target and (b) reference images from the contours of skirt masks (shown as insets) and then (c) transfer the texture of the reference to the target.

filter, for which we measured the difference between neighboring pixel intensities in the original image to avoid color leakage around the contour.

Figure 5 shows some results of automatic garment recoloring. The input images are those in the first and second rows in Fig. 4. We can see that the alpha mattes and recolored results obtained using our DA+PE masks are consistently better than the other results and are comparable to those obtained using ground truth.

### 6.2 Garment texture transfer

We have also implemented a simple application to automatically transfer the texture in a specific garment region in a reference image to a target image (see Fig. 6). We first generate alpha mattes in the same way as for recoloring. We then parameterize the contours of the binarized alpha mattes for the reference and target images, and calculate texture coordinates using mean value coordinates [28]. The warped texture is finally synthesized with the alpha matte of the target image. We keep the original shading in the target image by using an overlay.

### 6.3 Visualization for fashion analysis

We have used our human-image parsing results to visualize human-image data for fashion analysis, which enables a user to analyze fashion styles by mapping the human images into a 2D space. Specifically, we extract as features a normalized RGB histogram with 128 bins for each color channel from each image. In this process, multiple RGB histograms are individually computed from each region of $K$ types of garment ($0 \leqslant K \leqslant 17$) specified by the user. Next, we concatenate all RGB histograms to obtain a $128 \times 3 \times K$-vector for each image. To embed these high-dimensional features into 2D space, we use t-SNE [29]. Although such simple features suffice to obtain visually consistent results, we hope to consider more sophisticated features in future work.

Figure 7 shows visualization results for some of the test data. When we used the features of entire regions of the images (i.e., without any labels specified) as shown in Fig. 7(a), people with similar garments are not located nearby: their locations significantly depend on their backgrounds. In contrast, Fig. 7(b) demonstrates the effectiveness of using human-image parsing results: people are mapped in accordance with similarity of the selected garment (e.g., hat) regardless of the varying backgrounds. Moreover, the user can select multiple labels as shown in Fig. 7(c). In this example, the user selected three labels: pants, skirt, and u-cloth. We can see that the images are mainly grouped by the type of garment (pants and skirt). Additionally, images with a specific garment are arranged in accordance with its color. We can also analyze the combination of multiple garments, e.g., orange pants and pale u-cloth in the window illustrated in the figure. These results demonstrate that our human-image parsing method is effective for fashion style visualization.

We note that Simo-Serra and Ishikawa also visualized fashion style images by using their CNN-based features [30]. While their approach can roughly distinguish a human in the foreground and background, our approach can consider more detailed garment types obtained by human-image parsing, as demonstrated.

## 7 Conclusions and future work

In this paper, we have proposed a novel data augmentation method and a novel neural network that transfers pose estimation information to the human-image parsing domain. We have also provided comparisons with previous work and verified that the data augmentation method and pose estimation-based network are effective for human-image parsing. Although the proposed method improved accuracies for most classes, accuracies of certain classes with small regions (e.g., scarf) were low. In the future, we

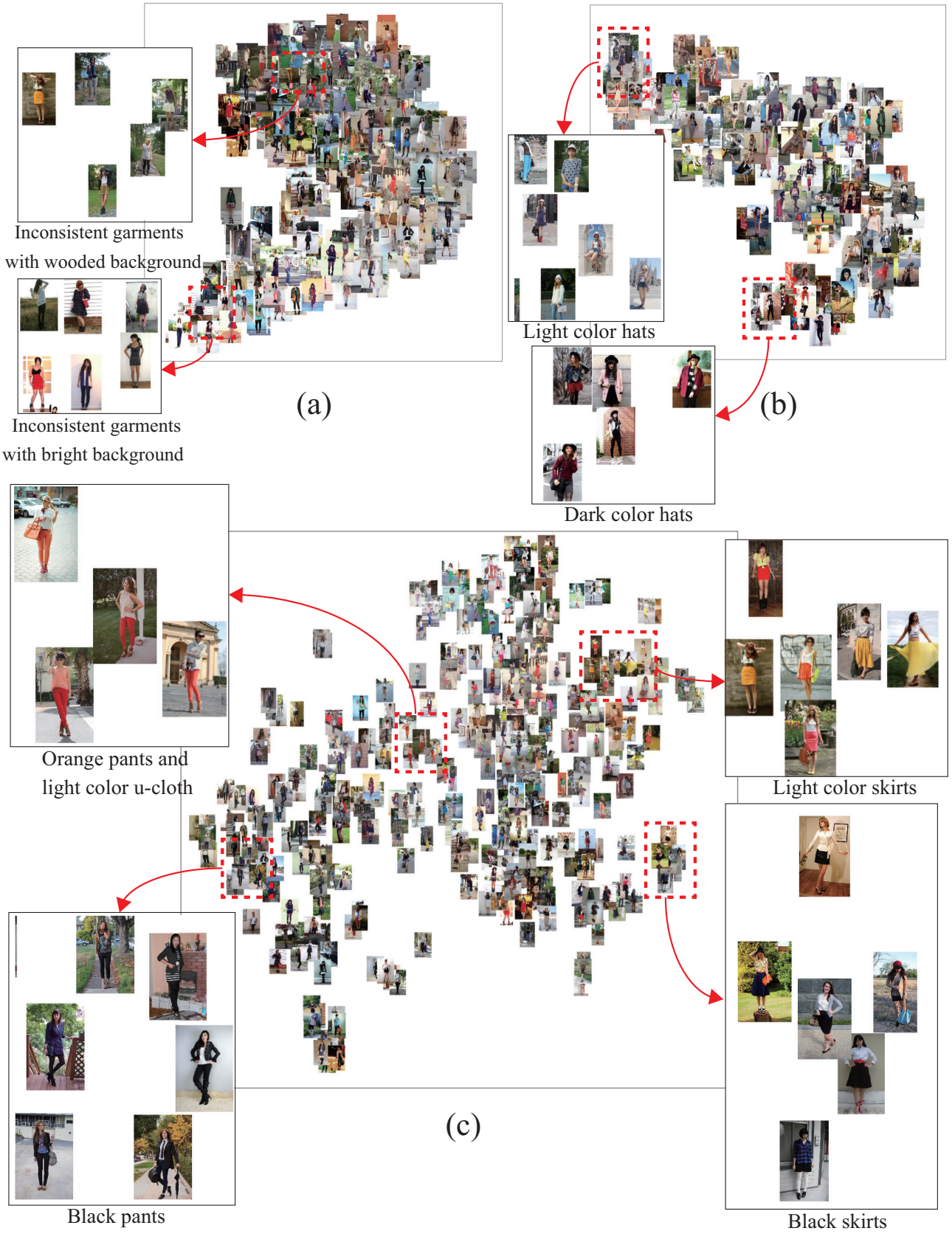



**Fig. 7** Visualization of human-image data for fashion style analysis by t-SNE [29], on the basis of features (a) from the entire region (i.e., without any labels specified), (b) with hat label, and (c) with pants, skirt, and u-cloth labels.

hope to improve performance for those classes with few training data. As done in Ref. [31], we would like to be able to deal with even less data by evenly sampling biased data.

**Electronic Supplementary Material** Supplementary material is available in the online version of this article at https://doi.org/s41095-017-0098-0.

## References


[1] Kanamori, Y.; Yamada, H.; Hirose, M.; Mitani, J.; Fukui, Y. Image-based virtual try-on system with garment reshaping and color correction. In: *Lecture Notes in Computer Science, Vol. 9550*. Gavrilova, M.; Tan, C.; Iglesias, A.; Shinya, M.; Galvez, A.; Sourin, A. Eds. Berlin, Heidelberg: Springer, 1–16, 2016.

[2] Di, W.; Wah, C.; Bhardwaj, A.; Piramuthu, R.; Sundaresan, N. Style finder: Fine-grained clothing style detection and retrieval. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 8–13, 2013.

[3] Hu, Y.; Yi, X.; Davis, L. S. Collaborative fashion recommendation: A functional tensor factorization approach. In: Proceedings of the 23rd ACM International Conference on Multimedia, 129–138, 2015.

[4] Kalantidis, Y.; Kennedy, L.; Li, L.-J. Getting the look: Clothing recognition and segmentation for automatic product suggestions in everyday photos. In: Proceedings of the 3rd ACM Conference on International Conference on Multimedia Retrieval, 105–112, 2013.

[5] Wei, S.-E.; Ramakrishna, V.; Kanade, T.; Sheikh, Y. Convolutional pose machines. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4724–4732, 2016.

[6] Liang, X.; Xu, C.; Shen, X.; Yang, J.; Tang, J.; Lin, L.; Yan, S. Human parsing with contextualized convolutional neural network. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 39, No. 1, 115–127, 2017.

[7] Quattoni, A.; Torralba, A. Recognizing indoor scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 413–420, 2009.

[8] Yamaguchi, K.; Kiapour, M. H.; Ortiz, L. E.; Berg, T. L. Parsing clothing in fashion photographs. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3570–3577, 2012.

[9] Yamaguchi, K.; Kiapour, M.; Ortiz, L.; Berg, T. Retrieving similar styles to parse clothing. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 37, No. 5, 1028–1040, 2015.

[10] Simo-Serra, E.; Fidler, S.; Moreno-Noguer, F.; Urtasun, R. A high performance CRF model for clothes parsing. In: Proceedings of the Asian Conference on Computer Vision, 64–81, 2014.

[11] Dong, J.; Chen, Q.; Shen, X.; Yang, J.; Yan, S. Towards unified human parsing and pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 843–850, 2014.

[12] Liu, S.; Liang, X.; Liu, L.; Lu, K.; Lin, L.; Yan, S. Fashion parsing with video context. In: Proceedings of the 22nd ACM International Conference on Multimedia, 467–476, 2014.

[13] Liang, X.; Liu, S.; Shen, X.; Yang, J.; Liu, L.; Dong, J.; Lin, L.; Yan, S. Deep human parsing with active template regression. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 37, No. 12, 2402–2414, 2015.

[14] Liu, S.; Liang, X.; Liu, L.; Shen, X.; Yang, J.; Xu, C.; Lin, L.; Cao, X.; Yan, S. Matching-CNN meets KNN: Quasi-parametric human parsing. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1419–1427, 2015.

[15] Bertasius, G.; Shi, J.; Torresani, L. Semantic segmentation with boundary neural fields. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3602–3610, 2016.

[16] Ghiasi, G.; Fowlkes, C. C. Laplacian pyramid reconstruction and refinement for semantic segmentation. In: Proceedings of the European Conference on Computer Vision, 519–534, 2016.

[17] Liang, X.; Shen, X.; Feng, J.; Lin, L.; Yan, S. Semantic object parsing with graph LSTM. In: Proceedings of the European Conference on Computer Vision, 125–143, 2016.

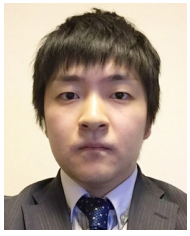
[18] Liang, X.; Shen, X.; Xiang, D.; Feng, J.; Lin, L.; Yan, S. Semantic object parsing with local-global long short-term memory. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3185–3193, 2016.

[19] Lin, G.; Shen, C.; van den Hengel, A.; Reid, I. Efficient piecewise training of deep structured models for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3194–3203, 2016.

[20] Vemulapalli, R.; Tuzel, O.; Liu, M.-Y.; Chellapa, R. Gaussian conditional random field network for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3224–3233, 2016.

[21] Dai, J.; He, K.; Sun, J. Instance-aware semantic segmentation via multi-task network cascades. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3150–3158, 2016.
[22] Hong, S.; Oh, J.; Lee, H.; Han, B. Learning transferrable knowledge for semantic segmentation with deep convolutional neural network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3204–3212, 2016.
[23] Papandreou, G.; Chen, L.; Murphy, K. P.; Yuille, A. L. Weakly- and semi-supervised learning of a deep convolutional network for semantic image segmentation. In: Proceedings of the IEEE International Conference on Computer Vision, 1742–1750, 2015.
[24] Yang, W.; Ouyang, W.; Li, H.; Wang, X. End-to-end learning of deformable mixture of parts and deep convolutional neural networks for human pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3073–3082, 2016.
[25] Chu, X.; Ouyang, W.; Li, H.; Wang, X. Structured feature learning for pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4715–4723, 2016.
[26] Andriluka, M.; Pishchulin, L.; Gehler, P.; Schiele, B. 2D human pose estimation: New benchmark and state of the art analysis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3686–3693, 2014.
[27] Aksoy, Y.; Aydin, T. O.; Pollefeys, M. Designing effective inter-pixel information flow for natural image matting. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 29–37, 2017.
[28] Floater, M. S. Mean value coordinates. *Computer Aided Geometric Design* Vol. 20, No. 1, 19–27, 2003.
[29] Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *Journal of Machine Learning Research* Vol. 9, 2579–2605, 2008.
[30] Simo-Serra, E.; Ishikawa, H. Fashion style in 128 floats: Joint ranking and classification using weak data for feature extraction. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 298–307, 2016.
[31] He, H.; Bai, Y.; Garcia, E. A.; Li, S. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In: Proceedings of the IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), 1322–1328, 2008.

**Takazumi Kikuchi** received his B.S. degree from the University of Tsukuba, Japan, in 2016. He is studying computer graphics and image processing on the master's course in computer science at the University of Tsukuba.



**Yuki Endo** received his B.S., M.S., and Ph.D. degrees in engineering from the University of Tsukuba, Japan, in 2010, 2012, and 2017, respectively. In 2016, he started working at the University of Tsukuba, where his present post is assistant professor in the Graduate School of Systems and Information Engineering. His research interests center on computer graphics and include image processing and machine learning.

**Yoshihiro Kanamori** received his B.S., M.S., and Ph.D. degrees in computer science from the University of Tokyo, Japan, in 2003, 2005, and 2009, respectively. He is an associate professor in the University of Tsukuba. He was a visiting researcher in ETH Zurich from 2014 to 2016 funded by the postdoctoral fellowship for research abroad of the Japan Society for the Promotion of Science (JSPS). His research interests center on computer graphics, especially rendering techniques. He studies image editing techniques for reproducing real-world phenomena as well as techniques for assisting creation of illustrations and animations.

**Taisuke Hashimoto** received his B.S. degree from the University of Tsukuba, Japan, in 2017. He is studying computer graphics and image processing on the master's course in computer science at the University of Tsukuba.

**Jun Mitani** received his Ph.D. degree in engineering from the University of Tokyo in 2004. He has been a professor at the University of Tsukuba since April 2015. His research interests center on computer graphics, in particular geometric modeling techniques and their application to curved origami as well as interactive design interfaces.