Retrieving Similar Styles to Parse Clothing

Kota Yamaguchi, *Member, IEEE,* M. Hadi Kiapour, *Student Member, IEEE,*
Luis E. Ortiz, and Tamara L. Berg, *Member, IEEE*

Abstract—Clothing recognition is a societally and commercially important yet extremely challenging problem due to large variations in clothing appearance, layering, style, and body shape and pose. In this paper, we tackle the clothing parsing problem using a retrieval-based approach. For a query image, we find similar styles from a large database of tagged fashion images and use these examples to recognize clothing items in the query. Our approach combines parsing from: pre-trained global clothing models, local clothing models learned on the fly from retrieved examples, and transferred parse-masks (Paper Doll item transfer) from retrieved examples. We evaluate our approach extensively and show significant improvements over previous state-of-the-art for both localization (clothing parsing given weak supervision in the form of tags) and detection (general clothing parsing). Our experimental results also indicate that the general pose estimation problem can benefit from clothing parsing.

Index Terms—Clothing parsing, clothing recognition, semantic segmentation, image parsing, pose estimation

1 Introduction

Clothing choices vary widely across the global population. For example, one person's style may lean toward preppy while another's trends toward goth. However, there are commonalities. For instance, walking through a college campus you might notice student after student consistently wearing combinations of jeans, t-shirts, sweatshirts, and sneakers. Or, you might observe those who have just stumbled out of bed and are wandering to class looking disheveled in their pajamas. Even hipsters who purport to be independent in their thinking and dress, tend to wear similar outfits consisting of variations on tight-fitting jeans, button down shirts, and thick plastic glasses. In some cases, style choices can be a strong cue for visual recognition.

In addition to style variation, individual clothing items also display many different appearance characteristics. As a concrete example, shirts have an incredibly wide range of appearances based on cut, color, material, and pattern. This can make identifying part of an outfit as a shirt very challenging. Luckily, for any particular choice of these parameters, *e.g.*, blue and white checked button down, there are many shirts with similar appearance. It is this visual similarity and the existence of some consistency in style choices discussed above that we exploit in our system.

In this paper, we take a data-driven approach to clothing parsing. We first collect a large, complex, real world collection of outfit pictures from a social network focused on fashion, chictopia.com. Using a small set of manually parsed images in combination with the text tags associated with each image in the collection, we can parse our large database accurately. Now, given a query image without any associated text, we can predict an accurate parse by retrieving similar outfits from our parsed collection, building local models from retrieved clothing items, and transferring inferred clothing items from the retrieved samples to the query image. Final iterative smoothing produces our end result. In each of these steps we take advantage of the relationship between clothing and body pose to constrain prediction and produce a more accurate parse. We call this *Paper Doll parsing* because it essentially transfers predictions from retrieved samples to the query, like laying paper cutouts of clothing items onto a paper doll. Consistencies in dressing make this retrieval-based effort possible.

In particular, we propose a retrieval-based approach to clothing parsing that combines:

- Pre-trained global models of clothing items
- Local models of clothing items learned on the fly from retrieved examples
- Parse mask predictions transferred from retrieved examples to the query image
- Iterative label smoothing

Previous state-of-the-art on clothing parsing [1] performed quite well for localization scenarios, where test images are parsed given user provided tags indicating depicted clothing items. However, this approach was less effective at unconstrained clothing parsing, where test images are parsed in the absence of any textual information (detection problem). In this paper, we use a large-scale dataset to solve the clothing parsing problem in this challenging detection scenario. An earlier version of this paper appeared at ICCV 2013 [2]. In this paper, we extend these initial ideas to provide extensive experiments exploring what factors contribute to per-

- *K. Yamaguchi is with the Graduate School of Information Sciences, Tohoku University, Sendai, Miyagi, Japan, 980-8579.*
- *L. Ortiz is with the Department of Computer Science, Stony Brook University, Stony Brook, NY, 11790.*
- *M. Kiapour and T. Berg are with the Department of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill, NC, 27599.*

formance improvements in our data-driven approach. Additionally, we provide new experiments evaluating how the resulting clothing parse can benefit the general pose estimation problem.

2 Related work

Clothing retrieval

There is a growing interest in clothing recognition problems, perhaps due to the huge potential for impact on e-commerce clothing applications (annual revenue for online shopping totals over $200 Billion dollars annually[1]). Automatic clothing recognition methods could enable natural and semantic image search for users of online fashion shops. This is reflected in the increasing number of papers related to clothing recognition for retrieval or recommendation applications [3], [4], [5], [6], [7], [8].

Most notably, the work of Liu et al. [3] proposes a street-to-shop application which tries to match pictures of clothing taken in the real world to clothing images in online shopping sites. In their approach, the authors consider a mapping between street and shopping images with a sparsely coded transfer matrix so that the difference between these two distributions does not affect the quality of retrieval. Kalantidis et al. [5] take a similar cross-scenario retrieval approach, where they utilize clothing parsing to explicitly represent each item. Cushen et al. [8] look at a similar problem, but with a focus on efficiency for mobile scenarios.

In addition to applications directly focused on clothing retrieval, clothing appearance similarity has been used for applications whose goal is to find the same person in similar poses across image or video collections [9].

As the interest in clothing-related applications grows, alongside these projects, there have been concerted efforts to create fashion-related datasets [1], [10], [11].

To enable clothing-related applications, we must be able (at some level) to recognize clothing in images. One way to do this is by clothing parsing where the goal is to predict a semantic label (e.g. shirt, pants, shoes) for each pixel on the person. The goal of our paper is to provide an approach for clothing parsing that could ultimately be used in a myriad of clothing applications.

Attribute recognition

Attributes of clothing are a natural way to describe its visual characteristics. For example, a user of an online shopping site might be looking for a “blue striped shirt” or a “red spectator pump”. In general attributes relate to visual characteristics of objects such as color, pattern, or shape. Attributes for clothing have been explored in several recent papers [6], [12], [13], [14], [15]. In general the attribute analysis is built upon detection and localization of items or parts of items in a picture.

The idea of clothing attribute recognition dates back to work by Borras et al. [16], which focused on recognizing clothing composites on upper-body detections. More recent work of Berg et al. proposes automatic attribute discovery and localization from shopping images using associated text description [12]. In related work, Bossard et al. provides methods for attribute classification in noisy Web images [15]. The work of Bourdev et al. [13] proposes the use of *poselets*, discriminative image patches that can capture small visual patterns in a picture, to detect clothing attributes such as “wearing hat”. Since attributes usually do not exist in isolation, Chen et al. considers co-occurrence between attributes during prediction using conditional random fields (CRF) [14].

One application of clothing attribute recognition is retrieval scenarios. Some work has been done in this area, using fine-grained attribute detection [6] or using human-in-the-loop approaches to interactively whittle down search results to what the user is looking for [17] or to build user specific models during search [18].

Clothing and person identification

Another important application of clothing recognition is the identification of people by their clothing. For example in temporal image collections, e.g. many pictures from an event, a person will be wearing the same clothes throughout the event. Therefore, clothing recognition is a strong cue to identity and has been used to recognize people in personal photo collections [19], [20], [21], repeated shots [22], or in surveillance scenarios [23], [24].

The clothing we wear is also a strong cue for predicting other characteristics about ourselves, such as social status, occupation, wealth, or occasion, to name a few. In this direction, there has been recent work on clothing recognition for occupation recognition [25], [26], fashion style recognition [27], or social tribe prediction in group photos [28], [29]. In the inverse direction, Liu et al. propose a system to recommend outfits (sets of clothing items) according to the occasion or event [4].

Clothing parsing

Clothing parsing is a relatively new computer vision task, but one that is important for enabling the above applications and for developing useful clothing representations. Early work on clothing representation modeled clothing as a grammar of sketch templates [30]. Other work took a subspace approach to describe clothing deformations [31], [32], or deformable spatial priors [33]. These approaches focus mainly on how to model shape deformations for clothing recognition.

We attack a somewhat different problem, that of clothing parsing. The clothing parsing problem was first formulated as an MAP estimation of superpixel labels using conditional random fields (CRF) [1]. The main insight of this method was the use of body pose estimation for clothing parsing. Dong et al. [34] later propose clothing parsing as an inference problem over *parselets*, a basis group of image regions that constitute clothing items. Liu et al. also propose a method to eliminate pixel-level supervision in learning using image-level color tags [35].

[1] http://blogs.wsj.com/digits/2012/02/27/online-retail-spending-at-200-billion-annually-and-growing/

Our approach differs in that 1) our method aims at recognition of fine-grained clothing categories without any prior information about an image, 2) our approach does not rely on over-segmentation, thus overcoming the limitation imposed by assuming uniformity in superpixels, and 3) our approach takes advantage of a large pool of freely available, weakly annotated Web images available in social networks focused on fashion.

Semantic segmentation

Clothing parsing is directly related to the well studied image parsing problem, where the goal is to assign a semantic object label to each pixel in an image. Most related to our paper are non-parametric methods for semantic segmentation [36], [37], [38] which have demonstrated state-of-the-art performance on the image parsing problem. Our approach shares the same non-parametric design for clothing parsing, but can additionally take advantage of pose estimates during parsing, and we do so in all parts of our method.

Pose estimation

Effective clothing parsing strongly depends on accurate human body pose localization. Therefore, we take advantage of recent advances in pose estimation [9], [39], [40], [41], [42]. Some previous clothing recognition work has used face detection to first find a person's head and torso and use this to bootstrap localization of given clothing items [33]. Also some approaches to the pose estimation problem itself have taken advantage of image segmentation [43], [44], [45] for improving performance. In this paper, we show empirical evidence that *semantic* clothing segmentation is beneficial to improving pose estimation.

3 Dataset

This paper uses the Fashionista dataset provided in [1] and a newly collected expansion called the Paper Doll dataset. The Fashionista dataset provides 685 images with clothing and pose annotation that we use for supervised training and performance evaluation, 456 for training and 229 for testing. The training samples are used to train a pose estimator, learn feature transformations, build global clothing models, and adjust parameters[2]. The testing samples are reserved solely for evaluation of both clothing parsing and pose estimation.

The Paper Doll dataset is a large collection of tagged fashion pictures with no manual annotation. We collected over 1 million pictures from chictopia.com with associated metadata tags denoting characteristics such as color, clothing item, or occasion. Since the Fashionista dataset was also collected from chictopia.com, we exclude any duplicate pictures from the Paper Doll dataset. From the remaining, we select pictures tagged with at least one item and run a full-body pose detector [40] that we learned from the Fashionista dataset, keeping those having a person detection. This results in 339,797 pictures weakly annotated with clothing items and estimated pose. Though the annotations are not always complete – users often do not label all of the items they are wearing, especially small items or accessories – it is rare to find images where an annotated tag is not present. We use the Paper Doll dataset for style retrieval.

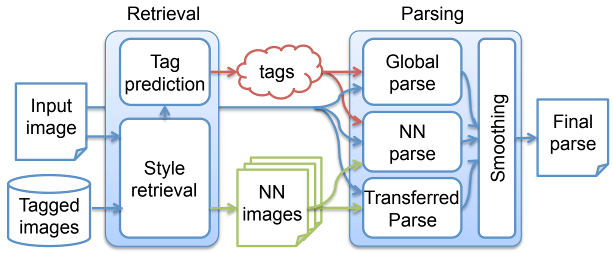


Fig. 1. Parsing pipeline. Retrieved images and predicted tags augment clothing parsing.

TABLE 1
Low-level features for parsing.

Name	Description
RGB	RGB color of the pixel.
Lab	L*a*b* color of the pixel.
MR8	Maximum Response Filters [47].
Gradients	Image gradients at the pixel.
HOG	HOG descriptor at the pixel [46].
Boundary Distance	Negative log-distance transform from the boundary of an image.
Pose Distance	Negative log-distance transform from 14 body joints and any body limbs.

4 Overview

Our parsing approach consists of two major steps:

- Retrieve similar images from the parsed database.
- Use retrieved images and tags to parse the query.

Figure 1 depicts the overall parsing pipeline. Section 5 describes our retrieval approach, and Section 6 details our parsing approach that combines three methods from the retrieval result.

Low-level features

We first run a pose estimator [40] and normalize the full-body bounding box to a fixed size, 302×142 pixels. The pose estimator is trained using the Fashionista training split and negative samples from the INRIA dataset [46]. During parsing, we compute the parse in this fixed frame size then warp it back to the original image, assuming regions outside the bounding box are background.

Our methods draw from a number of dense feature types (each parsing method uses some subset). Table 1 summarizes them.

[2] We learned parameters of the parsing model using pose estimation on the training images, where a pose estimator is learned from the same training images. This might incur a slight performance degradation in testing images, as the parsing model learns parameters from "cleaner" pose estimation. The mean average-precision-of-keypoints (APK) in train and test splits are 92.2% and 84.4%, respectively.

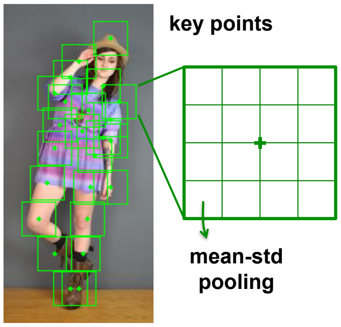


Fig. 2. Style descriptor: Compact representation for fashion images.

We compute Pose Distance by first interpolating 27 body joints estimated by a pose estimator [40] to obtain 14 points over body. Then, we compute a log-distance transform for each point. Also we compute log-distance transform of skeletal drawing of limbs (lines connecting 14 points). In total, we get a 15 dimensional vector for each pixel.

Whenever we use logistic regression [48] built upon these features in parsing, we first normalize features by subtracting their mean and dividing by 3 standard deviations for each dimension. Also, when we use logistic regression, we use these normalized features and their squares, along with a constant bias. So, for an N-dimensional feature vector, we always learn $2N + 1$ parameters. We find parameters of logistic regressions by 3-fold cross validation within training data.

5 Style Retrieval

Our goal for retrieving similar pictures is two-fold: a) to predict depicted clothing items, and b) to obtain information helpful for parsing clothing items.

5.1 Style descriptor

We design a descriptor for style retrieval that is useful for finding styles with similar appearance. For an image, we obtain a set of 24 key points interpolated from the 27 pose estimated body joints[3]. These key points are used to extract part-specific spatial descriptors - a mean-std pooling of normalized dense features in 4-by-4 cells in a 32-by-32 patch around the key point. That is, for each cell in the patch, we compute mean and standard deviation of the normalized features (Figure 2 illustrates). The features included in this descriptor are RGB, Lab, MR8, HOG, Boundary Distance, and Skin-hair Detection.

Skin-hair Detection is computed at each pixel using generalized logistic regression for 4 categories: *skin*, *hair*, *background*, and *clothing*. For the detector's input, we combine RGB, Lab, MR8, HOG, Boundary Distance, and Pose Distance to form a vector. Note that we do not include Pose Distance as a feature in the style descriptor, but instead use Skin-hair Detection to indirectly include pose-dependent information in the representation. We do this because the purpose of the style descriptor is to find similar styles robust to pose variation.

For each key point, we compute the above spatial descriptors and concatenate them to give a description of the overall style. This results in a 39,168 dimensional vector for each image. We use PCA to reduce dimensionality for efficiency of retrieval. We use the Fashionista training split to build the Skin-hair detector and also to train the PCA model. In our experiments, the descriptor resulted in 441 dimensional representation[4].

5.2 Retrieval

We use L2-distance over the style descriptors to find the K nearest neighbors (KNN) in the Paper Doll dataset. For efficiency, we build a KD-tree [49] to index samples. Unless noted, we fix $K = 25$ for all the experiments in this paper. Figure 3 shows two examples of nearest neighbor retrievals.

5.3 Tag prediction

The retrieved samples are first used to predict clothing items potentially present in a query image. The purpose of tag prediction is to obtain a set of tags that might be relevant to the query, while eliminating definitely irrelevant items for consideration. Later stages can remove spuriously predicted tags, but tags removed at this stage can never be predicted. Therefore, we wish to obtain the best possible performance in the high-recall regime.

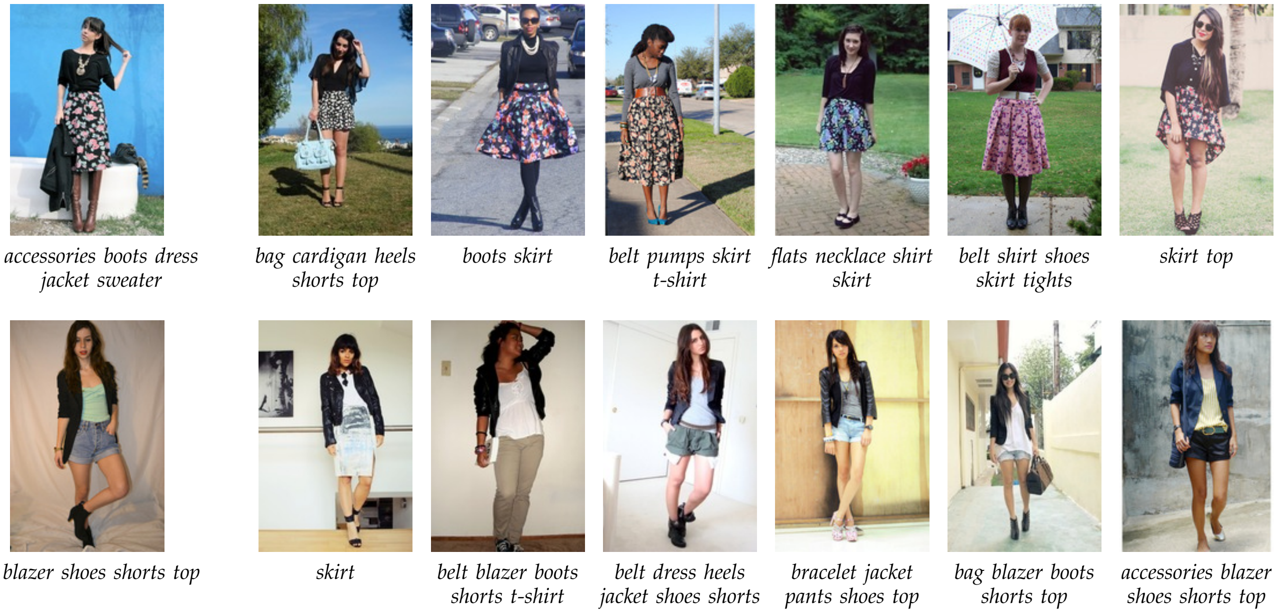
Our tag prediction is based on a simple voting approach from KNN. While simple, a data-driven approach is shown to be effective in tag prediction [50]. In our approach, each tag in the retrieved samples provides a vote weighted by the inverse of its distance from the query, which forms a confidence for presence of that item. We threshold this confidence to predict the presence of an item.

We experimentally selected this simple KNN prediction instead of other models because it turns out KNN works well for the high-recall prediction task. Figure 4 shows performance of linear vs KNN at 10 and 25. While linear classification (clothing item classifiers trained on subsets of body parts, e.g. *pants* on lower body keypoints), works well in the low-recall high-precision regime, KNN outperforms in the high-recall range. KNN at 25 also outperforms 10. The effect of retrieval size in parsing is evaluated in Section 7.2.

Since the goal here is only to eliminate obviously irrelevant items while keeping most potentially relevant items, we tune the threshold to give 0.5 recall in the Fashionista training split. Due to the skewed item distribution in the Fashionista dataset, we use the same threshold for all items to avoid over-fitting the prediction

[3] From the 27 point definition in Yang et al. [40], we removed point $4, 6, 16, 18$ and added interpolation of $(8, 20), (9, 21)$.

[4] We reduced dimensionality to account for 99% of variance in training data.



accessories boots dress jacket sweater

bag cardigan heels shorts top

boots skirt

belt pumps skirt t-shirt

flats necklace shirt skirt

belt shirt shoes skirt tights

skirt top

blazer shoes shorts top

skirt

belt blazer boots shorts t-shirt

belt dress heels jacket shoes shorts

bracelet jacket pants shoes top

bag blazer boots shorts top

accessories blazer shoes shorts top

Fig. 3. Retrieval examples. The leftmost column shows query images with ground truth item annotation. The rest are retrieved images with associated tags in the top 25. Notice retrieved samples sometimes have missing item tags.

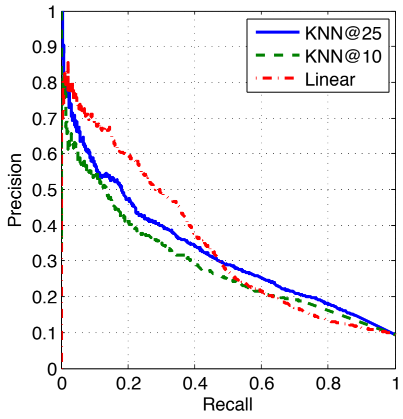


Fig. 4. Tag prediction PR-plot. KNN performs better in the high-recall regime.

model. In the parsing stage, we always include *background*, *skin*, and *hair* in addition to the predicted tags.

6 Clothing Parsing

Following tag prediction, we start to parse the image in a per-pixel fashion. Parsing has two major phases:

1) Compute pixel-level confidence from three methods: global parse, nearest neighbor parse, and transferred parse
2) Apply iterative label smoothing to get a final parse

Figure 5 illustrates outputs from each parsing stage.

6.1 Pixel confidence

We denote the clothing item label at pixel i by y_i. The first step is to compute a confidence score of assigning clothing item l to y_i. We model this scoring function S_Λ as the product mixture of three confidence functions.

$$\begin{aligned} S_\Lambda(y_i|\mathbf{x}_i, D) \quad \equiv \quad & S_{\text{global}}(y_i|\mathbf{x}_i, D)^{\lambda_1} \cdot \\ & S_{\text{nearest}}(y_i|\mathbf{x}_i, D)^{\lambda_2} \cdot \\ & S_{\text{transfer}}(y_i|\mathbf{x}_i, D)^{\lambda_3}, \end{aligned} \tag{1}$$

where we denote pixel features by $\mathbf{x}_i$, mixing parameters by $\Lambda \equiv [\lambda_1, \lambda_2, \lambda_3]$, and a set of nearest neighbor samples by D.

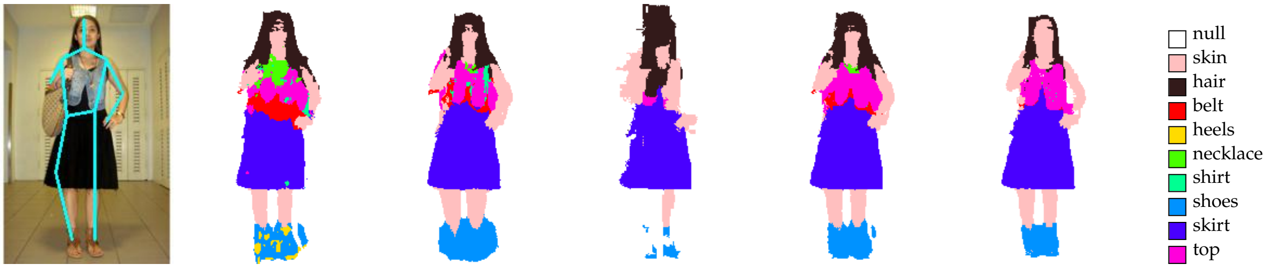
6.1.1 Global parse

The first term in our model is a global clothing likelihood, trained for each clothing item on the Fashionista training split. This is modeled as a logistic regression that computes the likelihood of a label assignment to each pixel for a given set of possible clothing items:

$$S_{\text{global}}(y_i|\mathbf{x}_i, D) \equiv P(y_i = l|\mathbf{x}_i, \theta_l^g) \cdot \mathbf{1}[l \in \tau(D)], \tag{2}$$

where P is logistic regression given feature $\mathbf{x}_i$ and model parameter θ_l^g, $\mathbf{1}[\cdot]$ is an indicator function, and $\tau(D)$ is a set of predicted tags from nearest neighbor retrieval. We use RGB, Lab, MR8, HOG, and Pose Distances as features. Any unpredicted items receive zero probability.

We trained the model parameter θ_l^g on the Fashionista training split. For training each θ_l^g, we select negative pixel samples only from those images having at least one positive pixel. That is, the model gives localization probability given that a label l is present in the picture. This could potentially increase confusion between similar item types, such as *blazer* and *jacket* since they usually do not appear together, in favor of better localization accuracy. We chose to rely on the tag prediction τ to resolve such confusion.



Pose estimation Global parse NN parse Transferred parse Combined Final result

Fig. 5. Parsing outputs at each step. Labels are MAP assignments of the scoring functions.

Because of the tremendous number of pixels in the dataset, we subsample pixels to train each of the logistic regression models. During subsampling, we try to sample pixels so that the resulting label distribution is close to uniform in each image, preventing learned models from only predicting large items.

6.1.2 *Nearest neighbor parse*

The second term in our model is also a logistic regression, but trained only on the retrieved nearest neighbor (NN) images. Here we learn a local appearance model for each clothing item based on examples that are similar to the query, e.g. *blazers* that look similar to the query blazer because they were retrieved via style similarity. These local models are much better models for the query image than those trained globally (because *blazers* in general can take on a huge range of appearances).

$$S_{\text{nearest}}(y_i|\mathbf{x}_i, D) \equiv P(y_i = l|\mathbf{x}_i, \theta_l^n) \cdot \mathbf{1}[l \in \tau(D)]. \qquad (3)$$

We learned the model parameter θ_l^n locally from the retrieved samples D, using RGB, Lab, Gradient, MR8, Boundary Distance, and Pose Distance.

In this step, we learn local appearance models using predicted pixel-level annotations from the retrieved samples computed during pre-processing detailed in Section 6.3. We train NN models using any pixel (with subsampling) in the retrieved samples in an one-vs-all fashion.

6.1.3 *Transferred parse*

The third term in our model is obtained by transferring the parse-mask likelihoods estimated by the global parse S_{global} from the retrieved images to the query image (Figure 6 depicts an example). This approach is similar in spirit to approaches for general segmentation that transfer likelihoods using over-segmentation and matching [51], [52], [53]; but here, because we are performing segmentation on people, we can take advantage of pose estimates during transfer.

In our approach, we find dense correspondence based on super-pixels instead of pixels (e.g., Tighe and Lazebnik [36]) to overcome the difficulty in naively transferring deformable, often occluded clothing items pixel-wise. Our approach first computes an over-segmentation of both query and retrieved images using a fast and simple segmentation algorithm [54], then finds corresponding pairs of super-pixels between the query and each retrieved image based on pose and appearance:

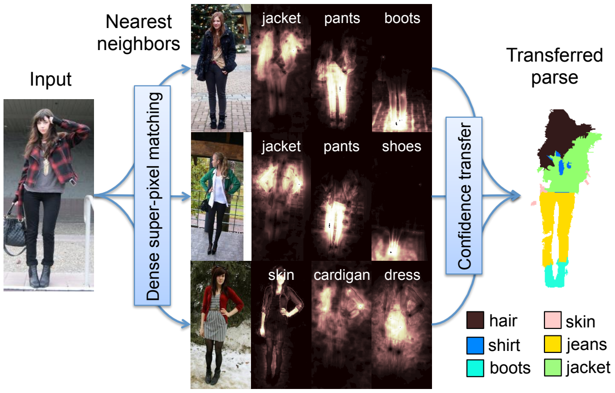


Fig. 6. Transferred parse. We transfer likelihoods in nearest neighbors to the input via dense matching.

1) For each super-pixel in the query, find the 5 nearest super-pixels in each retrieved image using L2 Pose Distance.
2) At each super-pixel, compute a bag-of-words representation [55] for each of RGB, Lab, MR8, and Gradient, and concatenate all.
3) Pick the closest super-pixel from each retrieved image using L2 distance on the concatenated bag-of-words feature.

Let us denote the super-pixel of pixel i by s_i, the selected corresponding super-pixel from image r by $s_{i,r}$, and the bag-of-words features of super-pixel s by $h(s)$. Then, we compute the transferred parse as

$$S_{\text{transfer}}(y_i|\mathbf{x}_i, D) \equiv \frac{1}{Z} \sum_{r \in D} \frac{M(y_i, s_{i,r})}{1 + \|h(s_i) - h(s_{i,r})\|}, \qquad (4)$$

where we define

$$M(y_i, s_{i,r}) \equiv \frac{1}{|s_{i,r}|} \sum_{j \in s_{i,r}} P(y_i = l|\mathbf{x}_i, \theta_l^g) \cdot \mathbf{1}[l \in \tau(r)], \qquad (5)$$

which is a mean of the global parse over the super-pixel in a retrieved image. Here we denote a set of tags of image r by $\tau(r)$, and the normalization constant by Z.

6.1.4 Combined confidence

After computing our three confidence scores, we combine them with parameter Λ to get the final pixel confidence S_Λ as described in Equation 1. We choose the best mixing parameter such that MAP assignment of pixel labels gives the best foreground accuracy in the Fashionista training split by solving the following optimization (on foreground pixels F):

$$\max_{\Lambda} \sum_{i \in F} \mathbf{1} \left[\tilde{y}_i = \arg\max_{y_i} S_\Lambda(y_i|\mathbf{x}_i) \right], \quad (6)$$

where $\tilde{y}_i$ is the ground truth annotation of the pixel i. For simplicity, we drop the nearest neighbors D in S_Λ from the notation. We use a simplex search algorithm over the simplex induced by the domain of Λ to solve for the optimum parameter starting from uniform values. In our experiment, we obtained $(0.41, 0.18, 0.39)$ using the training split of the Fashionista dataset.

We exclude background pixels from this optimization because of the skew in the label distribution – background pixels in Fashionista dataset represent 77% of total pixels, which tends to direct the optimizer to find meaningless local optima; i.e., predicting everything as *background*.

6.2 Iterative label smoothing

The combined confidence gives a rough estimate of item localization. However, it does not respect boundaries of actual clothing items since it is computed per-pixel. Therefore, we introduce an iterative smoothing stage that considers all pixels together to provide a smooth parse of an image. Following the approach of Shotton et al. [56], we formulate this smoothing problem by considering the joint labeling of pixels $Y \equiv \{y_i\}$ and item appearance models $\Theta \equiv \{\theta_l^s\}$, where θ_l^s is a model for a label l. The goal is to find the optimal joint assignment Y^* and item models Θ^* for a given image.

We start smoothing by initializing the current predicted parsing $\hat{Y}_0$ with the MAP assignment under the combined confidence S. Then, we treat $\hat{Y}_0$ as training data to build initial image-specific models $\hat{\Theta}_0$ (logistic regressions). We only use RGB, Lab, and Boundary Distance since otherwise models easily over-fit. Also, we use a higher regularization parameter for training instead of finding the best cross-validation parameter, assuming the initial training labels $\hat{Y}_0$ are noisy.

After obtaining $\hat{Y}_0$ and $\hat{\Theta}_0$, we solve for the optimal assignment $\hat{Y}_t$ at the current step t with the following optimization:

$$\hat{Y}_t \in \arg\max_{Y} \prod_i \Phi(y_i|\mathbf{x}_i, S, \hat{\Theta}_t) \prod_{i,j \in V} \Psi(y_i, y_j|\mathbf{x}_i, \mathbf{x}_j), \quad (7)$$

where we define:

$$\Phi(y_i|\mathbf{x}_i, S, \hat{\Theta}_t) \equiv \quad S(y_i|\mathbf{x}_i)^\lambda \cdot P(y_i|\mathbf{x}_i, \theta_l^s)^{1-\lambda}, \quad (8)$$
$$\Psi(y_i, y_j|\mathbf{x}_i, \mathbf{x}_j) \equiv \quad \exp\{\gamma e^{-\beta\|\mathbf{x}_i - \mathbf{x}_j\|^2} \cdot \mathbf{1}[y_i \neq y_j]\}. \quad (9)$$

Here, V is a set of neighboring pixel pairs, λ, β, γ are the parameters of the model, which we set to $\beta = -0.75, \lambda = 0.5, \gamma = 1.0$ in this paper according to perceptual quality in the training images[5]. We use the graph-cut algorithm [57], [58], [59] to find the optimal solution.

With the updated estimate of the labels $\hat{Y}_t$, we learn the logistic regressions $\hat{\Theta}_t$ and repeat until the algorithm converges. Note that this iterative approach is not guaranteed to converge. We terminate the iteration when 10 iterations pass, when the number of changes in label assignment is less than 100, or the ratio of the change is smaller than 5%.

6.3 Offline processing

Our retrieval techniques require the large Paper Doll dataset to be pre-processed (parsed), for building nearest neighbor models on the fly from retrieved samples and for transferring parse-masks. Therefore, we estimate a clothing parse for each sample in the 339K image dataset, making use of pose estimates and the tags associated with the image by the photo owner. This parse makes use of the global clothing models (constrained to the tags associated with the image by the photo owner) and iterative smoothing parts of our approach.

Although these training images are tagged, there are often clothing items missing in the annotation. This will lead iterative smoothing to mark foreground regions as *background*. To prevent this, we add an *unknown* item label with uniform probability and initialize $\hat{Y}_0$ together with the global clothing model at all samples. This effectively prevents the final estimated labeling $\hat{Y}$ to mark missing items with incorrect labels.

Offline processing of the entire Paper Doll dataset took a few days using our Matlab implementation in a distributed environment. For a novel query image, our full parsing pipeline takes 20 to 40 seconds, including pose estimation. The major computational bottlenecks are the nearest neighbor parse and iterative smoothing.

7 EXPERIMENTAL RESULTS

7.1 Parsing performance

We evaluate parsing performance on the 229 testing samples from the Fashionista dataset. The task is to predict a label for every pixel where labels represent a set of 56 different categories – a very large and challenging variety of clothing items.

Performance is measured in terms of standard metrics: accuracy, average precision, average recall, and average F-1 score over pixels. In addition, we also include foreground accuracy (See Eq. 6) as a measure of how accurately each method is at parsing foreground regions (those pixels on the body, not on the background). Note that the average measures are over non-empty labels after calculating pixel-based performance for each since some labels are not present in the test set. Since there are

[5] It is computationally prohibitive to optimize the parameters.

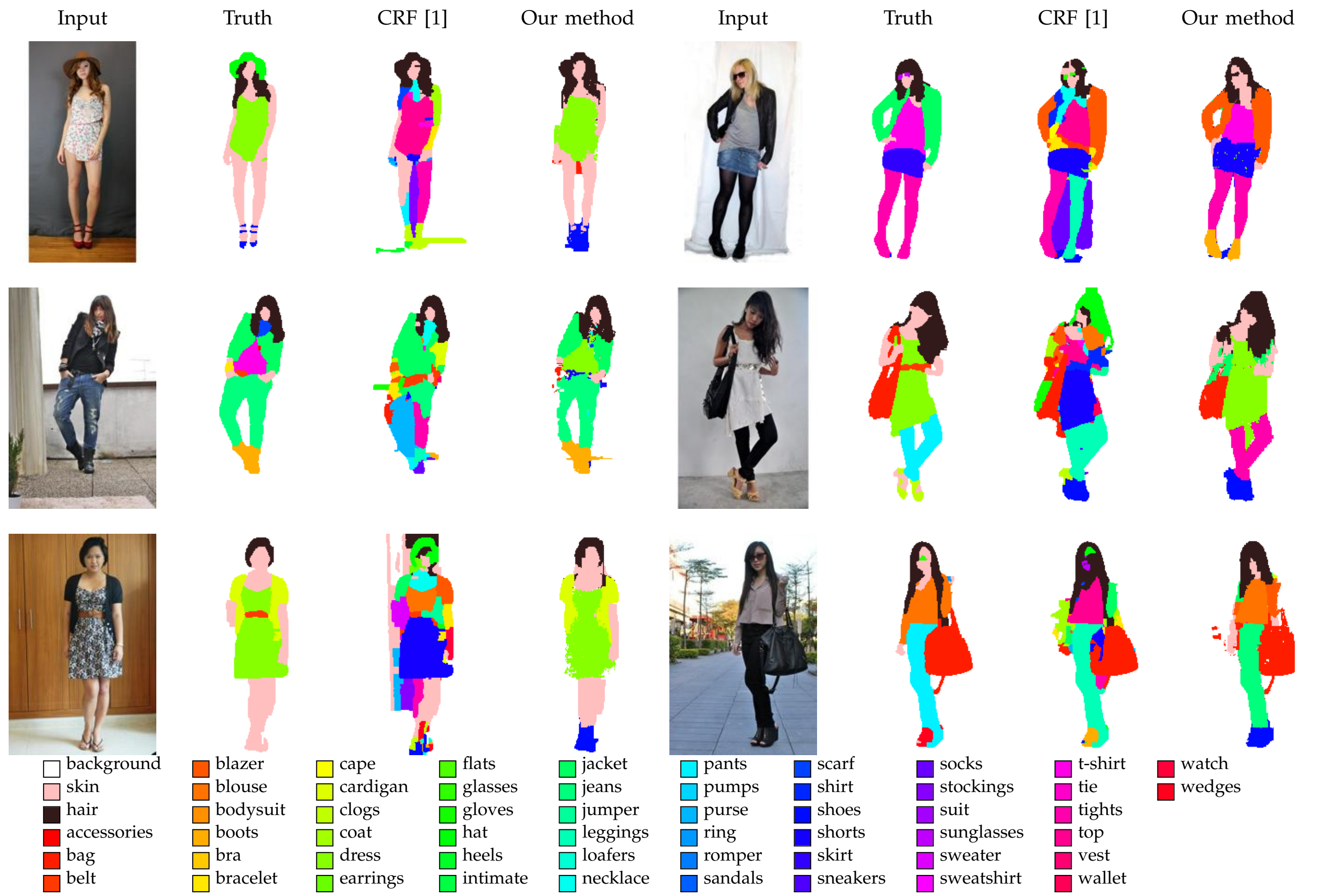


Fig. 7. Parsing examples (best seen in color). Our method sometimes confuses similar items, but gives overall perceptually better results.

TABLE 2
Parsing performance for final and intermediate results (MAP assignments at each step) in percentage.

Method	Accuracy	F.g. Accuracy	Avg. Precision	Avg. Recall	Avg. F-1
CRF [1]	77.45	23.11	10.53	**17.20**	10.35
Final	**84.68**	**40.20**	**33.34**	15.35	**14.87**
Global	79.63	35.88	18.59	15.18	12.98
Nearest	80.73	38.18	21.45	14.73	12.84
Transferred	83.06	33.20	31.47	12.24	11.85
Combined	83.01	39.55	25.84	15.53	14.22

some empty predictions, F-1 does not necessarily match the geometric mean of average precision and recall.

Table 2 summarizes predictive performance of our parsing method, including a breakdown of how well the intermediate parsing steps perform. For comparison, we include the performance of previous state-of-the-art on clothing parsing [1]. Our approach outperforms the previous method in overall accuracy (**84.68**% vs **77.45**%). It also provides a huge boost in foreground accuracy. The previous approach provides **23.11**% foreground accuracy, while we obtain **40.20**%. We also obtain much higher precision (**10.53**% vs **33.34**%) without much decrease in recall (**17.2**% vs **15.35**%).

In Table 2, we can observe that different parsing methods have different strength. For example, the global parse achieves higher recall than others, but the nearest-neighbor parse is better in foreground accuracy. Ultimately, we find that the combination of all three methods produces the best result. We provide further discussion in Section 7.4.

Figure 7 shows examples from our parsing method, compared to the ground truth annotation and the CRF-based method [1]. We observe that our method usually produces a parse that is qualitatively superior to the previous approach in that it usually respects the item boundary. In addition, many confusions are between similar item categories, e.g., predicting *pants* as *jeans*, or *jacket* as *blazer*. These confusions are reasonable due to high similarity in appearance between items and sometimes due to non-exclusivity in item types, i.e., *jeans* are a type of *pants*.

Figure 8 plots F-1 scores for non-empty items (items predicted on the test set) comparing the CRF-based method [1] with our method. Our model outperforms the previous work on many items, especially major foreground items such as *dress*, *jeans*, *coat*, *shorts*, or *skirt*.

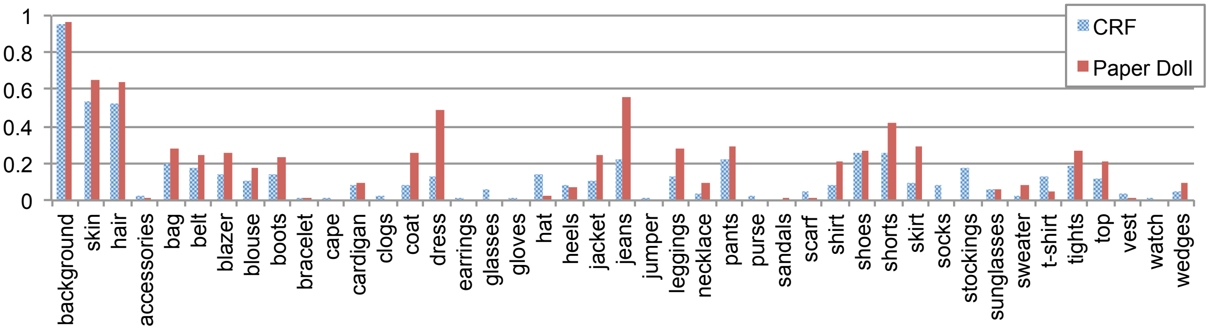


Fig. 8. F-1 score of non-empty items. We observe significant performance gains, especially for large items.

This results in a significant boost in foreground accuracy and perceptually better parsing results.

7.2 Big-data influence

To examine the performance of our proposed method in greater detail, we experiment with parsing performance as a function of retrieval set size in Figure 9 and as a function of data size in Figure 10.

Figure 9 shows the influence of the number of nearest neighbors retrieved on foreground accuracy, average precision, and average recall for each parsing stage. The plot also shows the previous CRF-based method [1] as a baseline. We observe a large performance boost between retrieval sets of size 1 image vs 2 images, which is mostly due to major missing items being filled in by the second nearest neighbor. Beyond that, the quality of tag prediction from the nearest neighbors gradually increases, resulting in performance improvement. In particular we see a large effect on average precision for larger retrieval set sizes. However, this performance increase comes with computational cost – In our implementation, it takes 8 seconds to parse one image if we retrieve only 1 image, but it takes 25 seconds if we retrieve 32 images. This is largely due to the increase in computational time to produce our NN parse and Transfer parse.

To understand how performance scales with data size, we examine parsing performance for random subsets of the Paper Doll dataset for varying data set size. For these experiments, we fix the retrieval set size to 25. The upper row in Figure 10 shows the performance plotted against the data size. We observe that all performance measures increase as the data size grow, but their rate differs; foreground accuracy and average recall show a moderate increase with respect to the data size, while average precision shows a major improvement for large data sizes. This result shows the benefit of big-data in our clothing parsing. Figure 11 shows examples of retrieval at data size = 256 and 262,144. Clearly, larger data size improves retrieval quality as well as predicted items.

These experiments show that our large-scale retrieval-based approach is effective for clothing parsing and that the size of the dataset is helpful for improved performance. The drawback of these kinds of approaches, is of course that it requires a fair amount of storage space. In our implementation, we use about 71GB of disk space to keep our preprocessed Paper Doll dataset. Also the performance improvement is proportional to the exponential growth of the data size. However, we emphasize that our approach does not require any manual annotation of the retrieval data set since we use only the tags already associated by the social network users to pre-parse this large external data set.

7.3 Localization vs. detection

The major motivation of using a retrieval-based approach is to do well on the *detection* problem – recognizing clothing in the absence of any supervision or labels. Another related problem is localization, where the depicted item labels are given and the goal is to localize these items in the corresponding picture. Both are interesting scenarios in different applications.

Our approach to parsing essentially has two parts. We predict candidate tags for an image given the retrieval set and localize those items in the image using our global, NN, and transferred parses. We have already examined performance of the first part in Figure 4. To determine an upper bound on the performance of the second part, we evaluate the performance of our method given a list of ground truth tags and compare that to performance of our full method (where candidate tags are predicted from the retrieval set). In the upper bound scenario, the global model is given the ground truth tags, the NN model only learns items included in the ground tags, and the transfer parse is not affected.

The second row of Figure 10 shows parsing performance over data size when depicted items are known before parsing. In this plot, we also report parsing performance when we apply iterative smoothing to the global parse (Global+Smooth) in addition to all intermediate results. Note that the CRF model [1] was specifically designed for the localization scenario and constitutes a strong baseline. Our final result performs better at average precision, with comparable result to the baseline in foreground accuracy and average recall. We also find that the most effective model for the localization scenario is our global model with iterative smoothing. Note that

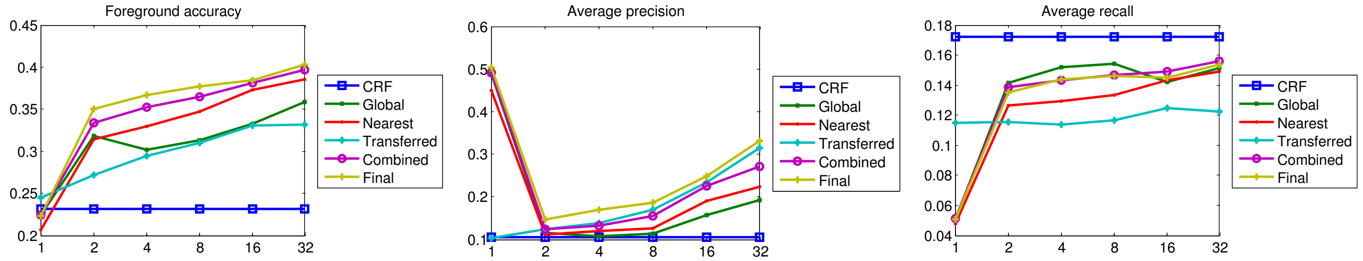


Fig. 9. Parsing performance over retrieval size when items are unknown. Larger retrieval size results in slightly better parsing, but also takes longer computation time.

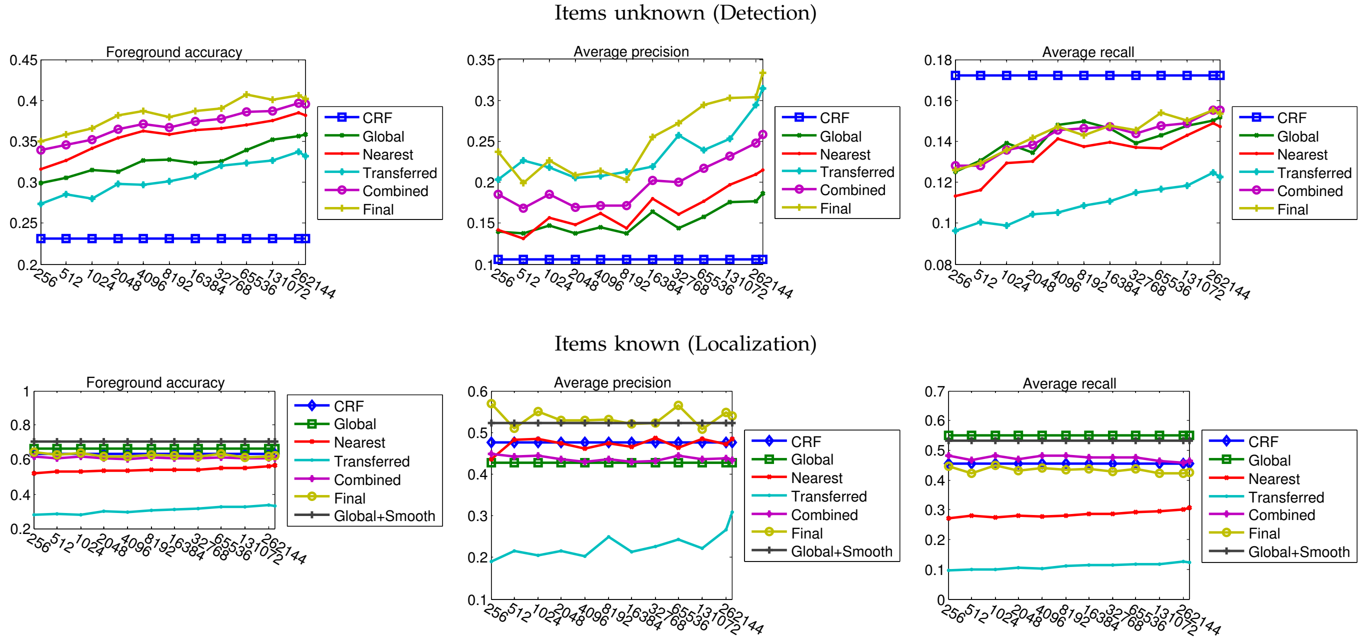


Fig. 10. Data size and parsing performance when 1) items are unknown and 2) items are known. While average recall tends to converge, average precision grows with data size.

this result outperforms the previous state-of-the-art in clothing localization [1], in all measures: foreground accuracy (**70.32% vs. 63.14%**), average precision (**52.24% vs. 47.63%**), and average recall (**53.25% vs. 45.54%**).

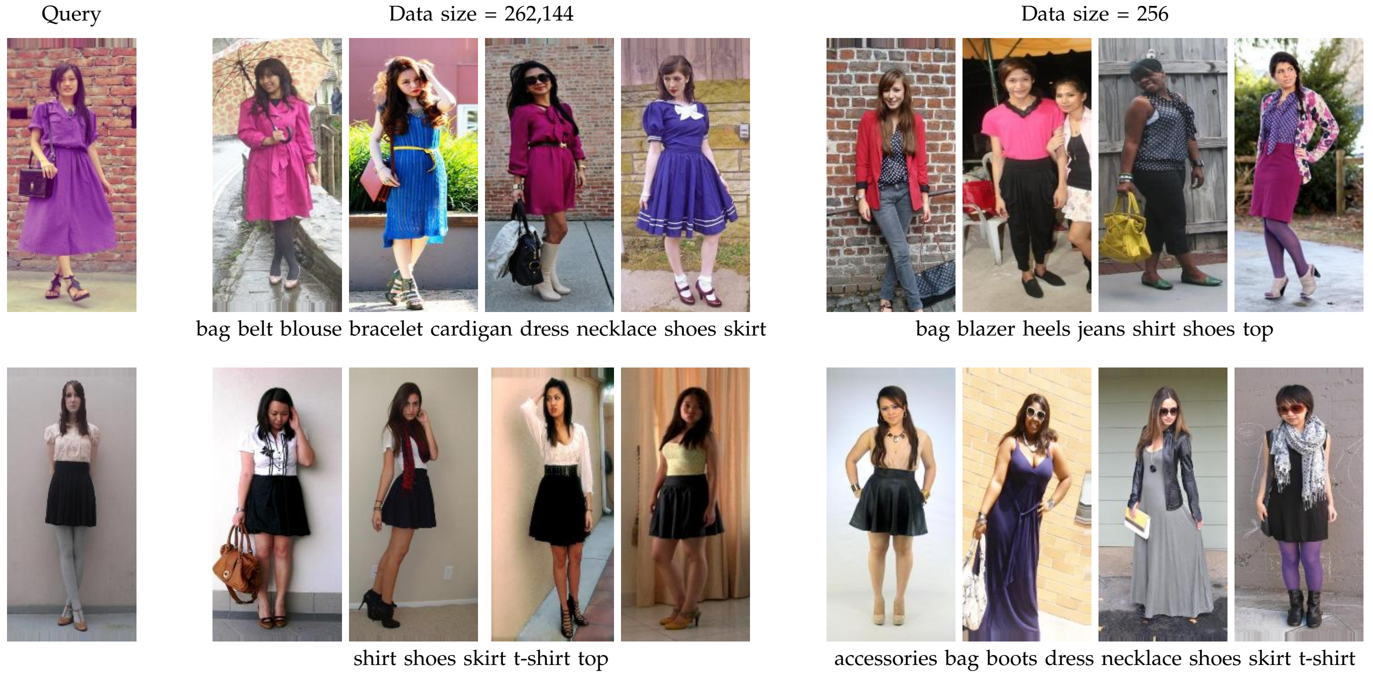
These results imply that localization performance is not strongly affected by retrieval. The result is expected, because the primary role of retrieval is to narrow down the list of potential items so that we reduce confusion in parsing. When items are known, the retrieval process no longer serves this role in parsing. Eventually, the global model is sufficient for producing a good result in the localization scenario. In other words, this result indicates that the big-data context is key to overcoming the performance gap between detection and localization.

7.4 Discussion

Though our method is successful at foreground prediction overall, there are a few drawbacks to our approach. By design, our style descriptor aims to represent whole outfit style rather than specific details of the outfit. Consequently, small items like accessories tend to be weighted less during retrieval and are therefore poorly predicted during parsing. This is also reflected in Table 2; the global parse is better than the nearest parse or the transferred parse in recall, because only the global parse could retain a stable appearance model of small items. However, in general, prediction of small items is inherently extremely challenging because of limited appearance information.

Another problem is the prevention of conflicting items from being predicted for the same image, such as *dress* and *skirt*, or *boots* and *shoes* which tend not to be worn together. Our iterative smoothing helps reduce such confusions, but the parsing result sometimes contains one item split into two conflicting items.

These two problems are the root of the error in tag prediction – either an item is missing or incorrectly predicted – and result in the performance gap between detection and localization. One way to resolve this would be to enforce constraints on the overall combination of



Query

Data size = 262,144

Data size = 256

bag belt blouse bracelet cardigan dress necklace shoes skirt

bag blazer heels jeans shirt shoes top

shirt shoes skirt t-shirt top

accessories bag boots dress necklace shoes skirt t-shirt

Fig. 11. Retrieval example for different data size. Predicted items are shown at the bottom. Notice at small data size, even a major item like dress or shirt can be missed in prediction.

predicted items, but this leads to a difficult optimization problem and we leave it as future work.

Lastly, we find it difficult to predict items with skin-like color or coarsely textured items. Because of the variation in lighting condition in pictures, it is very hard to distinguish between actual skin and clothing items that look like skin, e.g. slim khaki pants. Also, it is very challenging to differentiate for example between bold stripes and a belt using low-level image features. These cases will require higher-level knowledge about outfits to be correctly parsed.

Demo: An online demo of our parsing system is available at `clothingparsing.com`. Users can provide a url, image, or take a picture with their mobile device and view parsed results in 20-40 seconds. Users can also provide feedback about the quality of the results or directly edit the parsed results in the demo.

8 Parsing for pose estimation

In this section, we examine the effect of using clothing parsing to improve pose estimation. We take advantage of pose estimation in parsing, because clothing items are closely related to body parts. Similarly, we can benefit from clothing parsing in pose estimation, by using parsing as a contextual input in estimation.

We compare the performance of the pose estimator [40], using three different contextual input.

- **Baseline**: using only HOG feature at each part.
- **Clothing**: using a histogram of clothing in addition to HOG feature.
- **Foreground**: using a histogram of figure-ground segmentation in addition to HOG feature.

Here we concatenate all features into a single descriptor and learn a max-margin linear model [40]. All models use 5 mixture components in this experiment. We compute the foreground model simply by treating non-background regions in clothing parsing as foreground. Comparing the clothing model and the foreground model reveals how *semantic* information helps pose estimation compared to non-semantic segmentation. We use the Fashionista dataset in this comparison, with the same train-test split described in Section 3.

Table 3 summarizes average precision of keypoints (APK) and percentage of correct keypoints (PCK) using the Fashionista dataset. For clothing and foreground cases, we also compare the performance when we use the ground-truth pixel annotation, which serves as an upper bound on performance for each model given a perfect segmentation. Clearly, introducing clothing parsing improves the quality of pose estimation. Furthermore, the improvement of the clothing model over the foreground model indicates that the contribution is coming from the inclusion of *semantic* parsing rather than from a simple figure-ground segmentation.

From these performance numbers we can see that clothing parsing is particularly effective for improving localization of end parts of the body, such as the wrist. Perhaps this is due to items specific to certain body parts, such as *skin* for wrist and *shoes* for ankle. Note that a figure-ground segmentation cannot provide such semantic context. This result gives an important insight into the pose estimation problem, since improving esti-

TABLE 3
Pose estimation performance with or without conditional parsing input.

Average precision of keypoints (APK)

Method	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	Mean
Baseline	0.9956	0.9879	0.8882	0.5702	0.7908	0.8609	0.8149	0.8440
Clothing	1.0000	0.9927	0.8770	0.5601	0.8937	0.8868	0.8367	0.8639
- Ground truth	1.0000	0.9966	0.9119	0.6411	0.8658	0.9063	0.8586	0.8829
Foreground	1.0000	0.9926	0.8873	0.5441	0.8704	0.8522	0.7760	0.8461
- Ground truth	0.9976	0.9949	0.9244	0.5819	0.8527	0.8736	0.8118	0.8624

Percentage of correct keypoints (PCK)

Method	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	Mean
Baseline	0.9956	0.9891	0.9148	0.7031	0.8690	0.9017	0.8646	0.8911
Clothing	1.0000	0.9934	0.9127	0.6965	0.9345	0.9148	0.8843	0.9052
- Ground truth	1.0000	0.9978	0.9323	0.7467	0.9192	0.9367	0.9017	0.9192
Foreground	1.0000	0.9934	0.9148	0.6878	0.9127	0.8996	0.8450	0.8933
- Ground truth	0.9978	0.9956	0.9389	0.7183	0.9105	0.9214	0.8734	0.9080

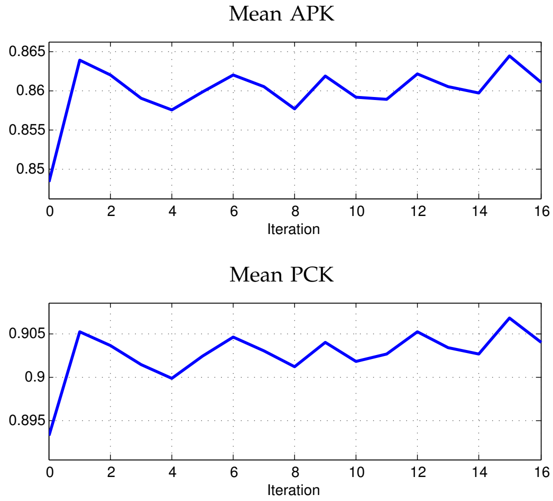


Fig. 12. Pose estimation performance over iterations.

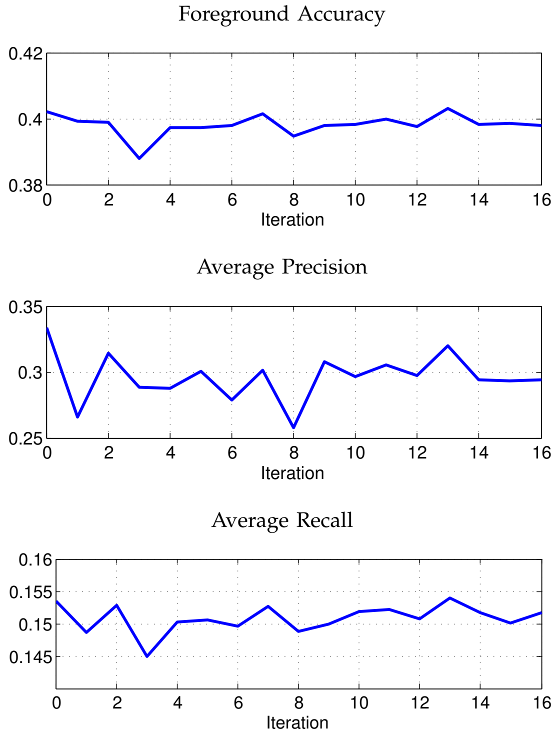


Fig. 13. Parsing performance over iterations.

mation quality for such end parts is the key challenge in pose estimation, while state-of-the-art methods can already accurately locate major parts such as head or torso. We believe that semantic parsing provides a strong context to improve localization of minor parts that often suffer from part articulation.

8.1 Iterating parsing and pose estimation

We have demonstrated that pose estimation can benefit from parsing. Since clothing parsing also depends on pose estimation, we also evaluate the effect of iterating between pose estimation and clothing parsing. This iterative process starts by clothing parsing with the baseline pose estimator, followed by pose estimation conditioned on clothing parsing. Then, we replace the pose estimation input to the parsing pipeline with the output of the conditional pose estimator, and continue the same process for several iterations. Denoting parsing with Y and pose configuration with Z, the process can be expressed for iteration $t = 0, 1, ..., n$:

$$Z_0 \equiv \arg\max_Z P(Z), \tag{10}$$

$$Y_t \equiv \arg\max_Y P(Y|Z_t), \tag{11}$$

$$Z_{t+1} \equiv \arg\max_Z P(Z|Y_t), \tag{12}$$

where $P(Z)$ is the baseline pose estimator, $P(Y|Z)$ is the parsing model, and $P(Z|Y)$ is the conditional pose estimator.

We evaluate the performance of pose estimation and parsing over iterations using the Fashionista dataset. Figure 12 and 13 plot the performance. The plot shows that the performance starts to oscillate after the first pose re-estimation by the conditional pose model. There is no clear benefit from repeated iterations in parsing for a few reasonable number of iterations, and the rate of improvement, if any, seems to be extremely slow. This result indicates that a slight change in pose estimation does not greatly affect the final parsing quality.

Oscillation happens because our model does not guarantee convergence. In this paper, we independently solve pose estimation and clothing parsing, and thus there is a discrepancy in the objective in this iterative process. To make the iterative approach converge, we need to consider a joint model of pose and parsing, and try to optimize for the global objective. Such an approach is an interesting future direction [45].

We conclude from this result that 1) the conditional pose estimator improves performance of pose re-estimation, but 2) there is little evidence that further iterations provide significant improved performance, and if anything, the rate of improvement is extremely slow.

9 Conclusion

We describe a clothing parsing method based on fashion image retrieval. Our system combines global parse models, nearest-neighbor parse models, and transferred parse predictions. Experimental evaluation shows successful results, demonstrating a significant boost of overall accuracy and especially foreground parsing accuracy over previous work in both detection and localization scenarios. The experimental results indicate that our data-driven approach is particularly beneficial in the detection scenario, where we need to both identify and localize clothing items without any prior knowledge about depicted clothing items. We also empirically show that pose estimation can benefit from the semantic information provided by clothing parsing.

Acknowledgements

Research supported by Google Faculty Award, ”Seeing Social: Exploiting Computer Vision in Online Communities,” NSF Career Award #1054133, and NSF Career Award #1054541.

References

[1] K. Yamaguchi, M. H. Kiapour, L. E. Ortiz, and T. L. Berg, “Parsing clothing in fashion photographs,” in *CVPR*, 2012, pp. 3570–3577.

[2] K. Yamaguchi, M. H. Kiapour, and T. L. Berg, “Paper doll parsing: retrieving similar styles to parse clothing items,” *ICCV*, 2013.

[3] S. Liu, Z. Song, G. Liu, C. Xu, H. Lu, and S. Yan, “Street-to-shop: Cross-scenario clothing retrieval via parts alignment and auxiliary set,” in *CVPR*, 2012, pp. 3330–3337.

[4] S. Liu, J. Feng, Z. Song, T. Zhang, H. Lu, C. Xu, and S. Yan, “Hi, magic closet, tell me what to wear!” in *ACM international conference on Multimedia*. ACM, 2012, pp. 619–628.

[5] Y. Kalantidis, L. Kennedy, and L.-J. Li, “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*. ACM, 2013, pp. 105–112.

[6] W. Di, C. Wah, A. Bhardwaj, R. Piramuthu, and N. Sundaresan, “Style finder: Fine-grained clothing style detection and retrieval,” in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on*. IEEE, 2013, pp. 8–13.

[7] M. Manfredi, C. Grana, S. Calderara, and R. Cucchiara, “A complete system for garment segmentation and color classification,” *Machine Vision and Applications*, pp. 1–15, 2013.

[8] G. A. Cushen and M. S. Nixon, “Mobile visual clothing search,” in *Multimedia and Expo Workshops (ICMEW), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1–6.

[9] V. Ferrari, M. Marin-Jimenez, and A. Zisserman, “Pose search: Retrieving people using their pose,” in *CVPR*, 2009.

[10] B. Loni, M. Menendez, M. Georgescu, L. Galli, C. Massari, I. S. Altingovde, D. Martinenghi, M. Melenhorst, R. Vliegendhart, and M. Larson, “Fashion-focused creative commons social dataset,” in *Proceedings of the 4th ACM Multimedia Systems Conference*. ACM, 2013, pp. 72–77.

[11] B. Loni, L. Y. Cheung, M. Riegler, A. Bozzon, L. Gottlieb, and M. Larson, “Fashion 10000: an enriched social image dataset for fashion and clothing,” in *Proceedings of the 5th ACM Multimedia Systems Conference*. ACM, 2014, pp. 41–46.

[12] T. L. Berg, A. C. Berg, and J. Shih, “Automatic attribute discovery and characterization from noisy web data,” in *Computer Vision–ECCV 2010*. Springer, 2010, pp. 663–676.

[13] L. Bourdev, S. Maji, and J. Malik, “Describing people: A poselet-based approach to attribute classification,” in *ICCV*, 2011, pp. 1543–1550.

[14] H. Chen, A. Gallagher, and B. Girod, “Describing clothing by semantic attributes,” in *ECCV*, 2012, pp. 609–623.

[15] L. Bossard, M. Dantone, C. Leistner, C. Wengert, T. Quack, and L. Van Gool, “Apparel classification with style,” *ACCV*, pp. 1–14, 2012.

[16] A. Borràs, F. Tous, J. Lladós, and M. Vanrell, “High-level clothes description based on colour-texture and structural features,” in *Pattern Recognition and Image Analysis*. Springer Berlin / Heidelberg, 2003, pp. 108–116.

[17] A. Kovashka, D. Parikh, and K. Grauman, “Whittlesearch: Image search with relative attribute feedback,” in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2973–2980.

[18] Q. Chen, G. Wang, and C. L. Tan, “Modeling fashion,” in *Multimedia and Expo (ICME), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1–6.

[19] D. Anguelov, K.-c. Lee, S. Gokturk, and B. Sumengen, “Contextual identity recognition in personal photo albums,” in *Computer Vision and Pattern Recognition, 2007. CVPR’07. IEEE Conference on*. IEEE, 2007, pp. 1–7.

[20] A. C. Gallagher and T. Chen, “Clothing cosegmentation for recognizing people,” in *CVPR*, 2008, pp. 1–8.

[21] N. Wang and H. Ai, “Who blocks who: Simultaneous clothing segmentation for grouping images,” in *ICCV*, 2011, pp. 1535–1542.

[22] J. Sivic, C. L. Zitnick, and R. Szeliski, “Finding people in repeated shots of the same scene,” in *BMVC*, 2006.

[23] M. Weber, M. Bäuml, and R. Stiefelhagen, “Part-based clothing segmentation for person retrieval,” in *AVSS*, 2011.

[24] M. Yang and K. Yu, “Real-time clothing recognition in surveillance videos,” in *ICIP*, 2011.

[25] Z. Song, M. Wang, X.-s. Hua, and S. Yan, “Predicting occupation via human clothing and contexts,” in *ICCV*, 2011, pp. 1084–1091.

[26] M. Shao, L. Li, and Y. Fu, “What do you do? occupation recognition in a photo via social context,” *ICCV*, 2013.

[27] M. H. Kiapour, K. Yamaguchi, A. C. Berg, and T. L. Berg, “Hipster wars: Discovering elements of fashion styles,” *ECCV*, 2014.

[28] A. C. Murillo, I. S. Kwak, L. Bourdev, D. Kriegman, and S. Belongie, “Urban tribes: Analyzing group photos from a social perspective,” in *CVPR Workshops*, 2012, pp. 28–35.

[29] I. S. Kwak, A. C. Murillo, P. N. Belhumeur, D. Kriegman, and S. Belongie, “From bikers to surfers: Visual recognition of urban tribes,” *BMVC*, 2013.

[30] H. Chen, Z. J. Xu, Z. Q. Liu, and S. C. Zhu, “Composite templates for cloth modeling and sketching,” in *CVPR*, 2006.

[31] A. O. Bălan and M. J. Black, “The naked truth: Estimating body shape under clothing,” *ECCV*, pp. 15–29, 2008. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-88688-4_2

[32] P. Guan, O. Freifeld, and M. J. Black, “A 2D human body model dressed in eigen clothing,” *ECCV*, pp. 285–298, 2010. [Online]. Available: http://dl.acm.org/citation.cfm?id=1886063.1886086

[33] B. Hasan and D. Hogg, "Segmentation using deformable spatial priors with application to clothing," in *BMVC*, 2010.

[34] J. Dong, Q. Chen, W. Xia, Z. Huang, and S. Yan, "A deformable mixture parsing model with parselets," *ICCV*, 2013.

[35] S. Liu, J. Feng, C. Domokos, H. Xu, J. Huang, Z. Hu, and S. Yan, "Fashion parsing with weak color-category labels," *IEEE Transactions on Multimedia*, vol. 16, no. 1, January 2014.

[36] J. Tighe and S. Lazebnik, "Superparsing: scalable nonparametric image parsing with superpixels," *ECCV*, pp. 352–365, 2010.

[37] C. Liu, J. Yuen, and A. Torralba, "Sift flow: Dense correspondence across scenes and its applications," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 5, pp. 978–994, 2011.

[38] J. Tighe and S. Lazebnik, "Finding things: Image parsing with regions and per-exemplar detectors," *CVPR*, 2013.

[39] D. Ramanan, "Learning to parse images of articulated bodies," in *NIPS*, 2006, pp. 1129–1136.

[40] Y. Yang and D. Ramanan, "Articulated pose estimation with flexible mixtures-of-parts," in *CVPR*, 2011, pp. 1385–1392.

[41] L. Bourdev, S. Maji, T. Brox, and J. Malik, "Detecting people using mutually consistent poselet activations," in *ECCV*, 2010. [Online]. Available: http://www.eecs.berkeley.edu/~lbourdev/poselets

[42] M. Dantone, J. Gall, C. Leistner, and L. Van Gool, "Human pose estimation using body parts dependent joint regressors," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013, pp. 3041–3048.

[43] P. Kohli, J. Rihan, M. Bray, and P. H. Torr, "Simultaneous segmentation and pose estimation of humans using dynamic graph cuts," *International Journal of Computer Vision*, vol. 79, no. 3, pp. 285–298, 2008.

[44] K. Alahari, G. Seguin, J. Sivic, I. Laptev *et al.*, "Pose estimation and segmentation of people in 3d movies," in *ICCV 2013-IEEE International Conference on Computer Vision*, 2013.

[45] L. Ladicky, P. H. Torr, and A. Zisserman, "Human pose estimation using a joint pixel-wise and part-wise formulation," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013, pp. 3578–3585.

[46] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, vol. 1. IEEE, 2005, pp. 886–893.

[47] M. Varma and A. Zisserman, "A statistical approach to texture classification from single images," *Int. J. Computer Vision*, vol. 62, no. 1-2, pp. 61–81, 2005.

[48] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *J Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.

[49] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," 2008. [Online]. Available: http://www.vlfeat.org/

[50] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid, "Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation," in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 309–316.

[51] E. Borenstein and J. Malik, "Shape guided object segmentation," in *CVPR*, vol. 1, 2006, pp. 969–976.

[52] B. Leibe, A. Leonardis, and B. Schiele, "Robust object detection with interleaved categorization and segmentation," *IJCV*, vol. 77, no. 1-3, pp. 259–289, 2008.

[53] M. Marszałek and C. Schmid, "Accurate object recognition with shape masks," *IJCV*, vol. 97, no. 2, pp. 191–209, 2012.

[54] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *IJCV*, vol. 59, no. 2, pp. 167–181, 2004.

[55] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*. IEEE, 2003, pp. 1470–1477.

[56] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation," *ECCV*, pp. 1–15, 2006.

[57] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, no. 11, pp. 1222–1239, 2001.

[58] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 9, pp. 1124–1137, 2004.

[59] V. Kolmogorov and R. Zabin, "What energy functions can be minimized via graph cuts?" *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 2, pp. 147–159, 2004.

Kota Yamaguchi Kota Yamaguchi is an Assistant Professor in the Graduate School of Information Sciences at Tohoku University, Japan. He received a PhD degree in Computer Science in 2014 from Stony Brook University. He received a MS in 2008 and a BEng in 2006, respectively, both from the University of Tokyo. His research interest is in computer vision and machine learning, with a focus on semantic understanding problems.

M. Hadi Kiapour Hadi Kiapour is currently a PhD student in Computer Science at University of North Carolina at Chapel Hill. His primary research interests lie in computer vision and machine learning with a focus on image understanding at Web-scale. Hadi was a visiting researcher at Computer Vision and Active Perception Lab (CVAP) at KTH University in Stockholm in 2010. He received a BS degree in electrical engineering from Sharif University of Technology in 2011.

Luis E. Ortiz Luis E. Ortiz is an Assistant Professor at Stony Brook University. Prior to joining Stony Brook, he was an Assistant Professor at the University of Puerto Rico, Mayagüez; a Postdoctoral Lecturer at MIT; a Postdoctoral Researcher at Penn; and a Consultant in the Field of AI and ML at AT&T LabsResearch. He received a MS and completed a PhD in Computer Science in 1998 and 2001, respectively, both from Brown. He received a BS in Computer Science from the University of Minnesota. His main research areas are AI and ML. His current focus is on computational game theory and economics, with applications to the study of influence in strategic, networked, large-population settings, and learning game-theoretic models from data on strategic behavior. Other interests include, game-theoretic models for interdependent security, algorithms for computing equilibria in games, connections to probabilistic graphical models, and AdaBoost.

Tamara L. Berg Tamara L. Berg completed her PhD degree in Computer Science at U.C. Berkeley in 2007. She is currently an Assistant Professor at UNC Chapel Hill. Prior to that, Tamara was an Assistant Professor at Stony Brook University from 2008-2013 and a research scientist at Yahoo! Research from 2007-2008. She was also winner of the Marr Prize in 2013. Tamara's main research focus is on integrating disparate information from text and image analysis. Other areas of interest include integrating computational recognition with social analysis, especially for style and trend prediction. She holds a BS in Mathematics and Computer Science from the University of Wisconsin, Madison.