Where and Who?
Automatic Semantic-Aware Person Composition

FUWEN TAN, University of Virginia
CRISPIN BERNIER, University of Virginia
BENJAMIN COHEN, University of Virginia
VICENTE ORDONEZ, University of Virginia
CONNELLY BARNES, University of Virginia

Fig. 1. Given an input image (A, C), the proposed system can automatically composite a person instance on the background. The composite looks semantically convincing and visually pleasing (B, D). In order to achieve this, the system first predicts the location and size of the potential segment by analyzing the scene semantics. A favorable segment is then retrieved from a candidate pool and composited on the background automatically.

Image compositing is a popular and successful method used to generate realistic yet fake imagery. Much previous work in compositing has focused on improving the appearance compatibility between a given object segment and a background image. However, most previous work does not investigate the topic of automatically selecting semantically compatible segments and predicting their locations and sizes given a background image. In this work, we attempt to fill this gap by developing a fully automatic compositing system that learns this information. To simplify the task, we restrict our problem by focusing on human instance composition, because human segments exhibit strong correlations with the background scene and are easy to collect. The first problem we investigate is determining where should a person segment be placed given a background image, and what should be its size in the background image. We tackle this by developing a novel Convolutional Neural Network (CNN) model that jointly predicts the potential location and size of the person segment. The second problem we investigate is, given the background image, which person segments (who) can be composited with the previously predicted locations and sizes, while retaining compatibility with both the local context and the global scene semantics? To achieve this, we propose an efficient context-based segment retrieval method that incorporates pre-trained deep feature representations.

To demonstrate the effectiveness of the proposed compositing system, we conduct quantitative and qualitative experiments including a user study. Experimental results show our system can generate composite images that look semantically and visually convincing. We also develop a proof-of-concept user interface to demonstrate the potential application of our method.

CCS Concepts: • **Computing methodologies → Image manipulation**; • **Applied computing → Image composition**;

2017. XXXX-XXXX/2017/6-ART1 $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

ACM Reference format:
Fuwen Tan, Crispin Bernier, Benjamin Cohen, Vicente Ordonez, and Connelly Barnes. 2017. Where and Who? Automatic Semantic-Aware Person Composition. 1, 1, Article 1 (June 2017), 11 pages.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Image compositing can produce images that can trick humans into believing that they are real, even though they are not. Image composites can also result in fantastic images that are limited only by the artist's imagination. However, the process of creating composite images is challenging, and it is not fully understood how to make realistic composites.

A typical compositing task proceeds in four steps: (1) choose a foreground segment that is semantically compatible with the background scene; (2) place the segment at a proper location with the right size; (3) perform operations such as alpha matting [Smith and Blinn 1996] or Poisson blending [Pérez et al. 2003] to adjust the local appearance; (4) apply global refinements such as relighting or harmonization [Tsai et al. 2017]. The first two steps require semantic reasoning while the last two steps help with appearance compatibility. Whether a human perceives a composite image as real or fake depends on all these factors. However, while existing compositing systems have made considerable efforts to tackle the last two steps automatically, most of them leave the semantic tasks (steps 1 and 2) to the users.

In this work, we explore the semantic relationships between a collection of foreground segments and background scenes using a data-driven method for automatic composition. We restrict the

foreground category to “human" because humans play a central role in a large proportion of image composites. By focusing on the “human" category, we can also easily collect enough exemplar data for training and testing our system. As a first attempt, we also choose to ignore occlusion by assuming that the human segments we composite are fully visible from the camera viewpoint.

This research is motivated by recent breakthroughs in scene recognition [Xiao et al. 2016] and object-level reasoning [Ren et al. 2015] through deep neural networks, which have brought unprecedented levels of performance for similar semantic tasks. Thus, we apply these techniques to estimate the semantic compatibility between candidate foreground segments and background images using a large scale visual dataset consisting of real image layouts and their context. Given these observations, our method contains three components:

First, using a background image as input, we extract the scene layout using a state-of-the-art object detection system [Ren et al. 2015]. Our system then predicts the location and size of each potential person instance, which is represented as a bounding box (4D vector), by a novel Convolutional Neural Network (CNN) model trained on large numbers of real examples. Direct regression in a four dimensional continuous space is difficult. Our key insight here is that by discretizing the bounding box space, we can formulate the prediction as a joint classification problem using a two branch cascade network, where the first branch predicts the location and the second branch predicts the size.

Second, once we obtain the potential bounding box of the person segment, our next challenge is to retrieve a proper segment that semantically matches the local context and the global scene. Fortunately, it turns out that the strong, consistent and regular correlation between objects and contexts, especially between persons and their surrounding scenes, can be successfully measured in feature space. Specifically, we develop a context based segment retrieval scheme using a pre-trained hybrid feature representation that incorporates both the local contextual cues and global scene semantics. We find that this simple but efficient design works surprisingly well.

Lastly, with the retrieved segment placed at the predicted location with the predicted size, we leverage off-the-shelf alpha matting [Chen et al. 2013] to adjust the transition between the composited segment and its surroundings so that the segment appears compatible with the background. Because general image segmentation is still a challenging task, when preparing the candidate person segments, for the highest quality results, we currently create an initial binary mask using manual segmentation with the Magnetic Lasso Tool from Adobe Photoshop. Fortunately, we only need to do this once for each segment and can precompute a candidate segment pool in advance.

To evaluate our method, we develop an automatic person compositing system and conduct quantitative and qualitative experiments including a user study. We also demonstrate that this generation pipeline can be useful for interactive layout design or storyboarding tasks which could not be easily fulfilled using other tools.

We summarize here our technical contributions: (1) We propose a novel CNN based model to directly predict the location and size of the potential foreground person segment; (2) We develop a context based segment retrieval scheme by incorporating deep feature representation which encodes both local contextual cues and global scene semantics; (3) We build an automatic person compositing system which generates convincing composite images. To the best of our knowledge, this is the first attempt towards this task; (4) We conduct quantitative and qualitative evaluations, including a user study and a proof-of-concept user interface to demonstrate the capacity of our system.

2 RELATED WORK

Composite image generation. As one of the most common approaches to create realistic images, composite image generation has been an active research topic in both the computer graphics and vision communities.

Early methods such as alpha matting [Smith and Blinn 1996] and gradient-domain compositing [Pérez et al. 2003] can seamlessly stitch a foreground object with a background image by blending the local transition region.

To enforce global appearance compatibility, Lalonde and Efros [2007] proposed to model the co-occurrence probability of the foreground object and the background image using a color distribution. Similarly, Xue et al. [2012] proposed to investigate the key statistical properties that control the realism of an image composite. Recently, Zhu et al. [2015] trained a single CNN based model to distinguish composite images from natural photographs and refine them by optimizing the predicted scores. Furthermore, Tsai et al. [2017] developed an end-to-end deep CNN based model for image harmonization. While these methods presented visually-pleasing results, they all leave the semantic tasks to the users, such as choosing the foreground segments and placing them at proper positions with the right size.

On the other hand, Lalonde et al. [2007] built an interactive system to insert new objects into existing photographs by querying a vast image-based object library. Chen et al. [2009] developed a similar interactive system but took user sketches as input. Hays et al. [2007] proposed an automatic patch retrieval and blending method for scene completion using millions of photographs. However, these methods still relied on hand-crafted features and the composite regions were indicated manually by the users.

Context based scene reasoning. Using context for scene reasoning has a long history [Divvala et al. 2009]. Pioneering works include Bar and Ullman [1991] and Strat and Fischler [1996], which incorporated contextual information for recognition. Context based methods also popularize in object-level classification. Bell et al. [2016] proposed a Recurrent Neural Network (RNN) framework to detect objects in context. All these works modeled correlations among contents within the image, while our method intends to predict contents that are not yet present. Related to our work, Torralba et al. [2003] introduced a context challenge to test to what extent can object detection succeed by exclusively contextual cues. Most recently, Sun et al. [2017] proposed a convolutional siamese network to detect missing objects in an image. While these methods predicted contents that were not present in the images, they all focused on the binary determination of whether there should be any object presented at a specific location or not. On the other hand, our method

attempts to predict both the location and size of a potential foreground segment, and also retrieve a segment with proper appearance that is compatible with the surrounding context. The work of Malisiewicz et al. [2009] is also relevant to ours, as they presented an exemplar based object retrieval method with a graph model which encoded the correlation among object instances in the scene. However, this method relied on hand-crafted features and did not predict the location and size of the object, nor did it target compositing.

Context based image editing. By conditioning on the local surroundings, Pathak et al. [2016] performed unsupervised visual feature learning and semantic inpainting by pixel-level regression. Yang et al. [2017] proposed a multi-scale CNN model for high-resolution image inpainting using neural patch synthesis. Most recently, Iizuka et al. [2017] developed a CNN based method for image completion by enforcing global and local consistency simultaneously. While all these methods tried to reconstruct the missing contents from context at the pixel level, the inpainted regions were still manually indicated by the users. Our method attempts to predict such regions and retrieve proper segments in feature space.

3 OVERVIEW

We show in Figure 2 an overview of our system. Our system has three main components: bounding box prediction, person segment retrieval, and compositing. We now give a brief discussion of each of these.

In Section 4, we introduce our proposed CNN based model to predict a bounding box of the potential segment. Data preprocessing is first applied to a large pool of person segments to exclude unsuitable ones such as those that are heavily occluded or too small. We then formulate the bounding box prediction as a joint classification problem by discretizing the spatial and size domains of the bounding box space. Specifically, we design a novel two branch network which can be trained end-to-end using supervised learning, and tested in a cascade manner.

In Section 5, we introduce the candidate pool we build for segment retrieval. Given a bounding box predicted using our method from Section 4, a context based segment retrieval scheme is then introduced to find a person segment from the candidate pool that semantically matches both the local context and the global scene. The key component for achieving this is a hybrid deep feature representation. Finally, we use an off-the-shelf alpha matting technique to seamlessly composite the retrieved segment with the background at the predicted location and size.

In Section 6, we evaluate our bounding box prediction model quantitatively by measuring the histogram correlation between the ground truth bounding boxes distribution and our prediction. We also evaluate the visual realism of composite images with a human subject evaluation.

4 BOUNDING BOX PREDICTION

In this section, we introduce our learning based method to predict the bounding box of a potential person segment given a single background image. Our key insight here is that the correlation between the foreground segment and the background scene can be learned directly from human-annotated object layouts of natural images.

We first discuss how we collect and preprocess the data (Section 4.1). Next, we explain the input for the learning model (Section 4.2), and give the prediction target for the model (Section 4.3). We then give the model itself (Section 4.4) and provide some implementation details (Section 4.5).

4.1 Data preprocessing

The data we use for learning such layout correlation is from the MS-COCO [Lin et al. 2014] dataset. This dataset contains tens of thousands of images with both bounding box and segment annotations for each object instance in 80 different categories, such as person, car, bus, etc.

Because a large proportion of object instances are occluded, we automatically filter out heavily occluded person instances using three passes of filtering: (1) We filter the person instances whose bounding boxes have large overlapping areas with other objects. Specifically, we exclude instances whose Intersection over Union (IoU) with any other instance is larger than 0.3. (2) We also exclude person instances that are close to the edge of the images as they are probably incomplete. In particular, we filter the instance if the distance between its bounding box and the edge of the image is less than 18 pixels. (3) Finally, we remove instances whose areas are less than 2500 square pixels.

After applying the filtering routines, we obtain 36,636 person instances from the training split of MS-COCO, and 16,962 from the validation split.

4.2 Input imagery

For each person instance in the dataset, we attempt to learn the mapping from its background context to the person's bounding box. Learning such a mapping function requires us having an input image in which the person is not already present. However, we cannot easily create such inputs from natural photographs unless we could perfectly "erase" the person instances from the source images. Our solution is to remove the person instances automatically by using the human-annotated segments from MS-COCO. We remove each person via the inpainting method of Barnes et al. [2009], implemented as Content Aware Fill from Adobe Photoshop. However, if the image belongs to a complex scene, the inpainted results may inevitably exhibit artifacts such as blurring or repetitive patches. To prevent the model from over-fitting on these artifacts, the inpainted image is further blurred using a Gaussian filtering with a sigma scale of 3.2. We denote the blurred image as I_B. An example blurred image is shown in Figure 3.

Given the recent breakthroughs in CNN-based object detection systems [Bell et al. 2016; Ren et al. 2015], in addition to using our inpainted (and blurred) images directly as input, we also incorporate the informative output from an object detector. We use the Faster RCNN object detector [Ren et al. 2015], which is pretrained on the MS-COCO dataset, to obtain object detections in the inpainted images. The bounding boxes of the detected objects in different categories are then rendered using a randomly generated color palette, with each color corresponding to a given category. The color values within an overlapping region are the mean color value of the corresponding instances. We find that using different color

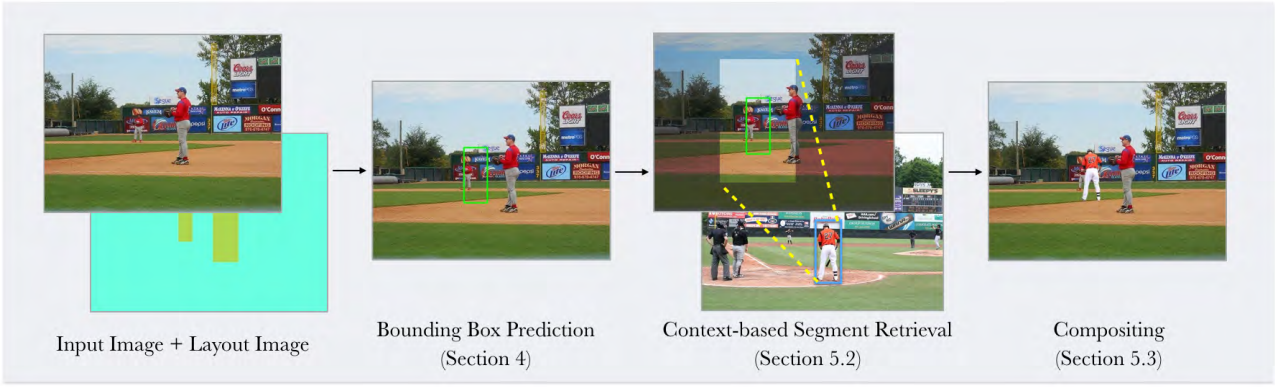


Fig. 2. Overview of our system: it consists of three computational stages (see above), from bounding box prediction to final compositing.

Fig. 3. Input data preparation: (A) input image, (B) inpainted image with the "man in red coat" erased, (C) blurred image, (D) rendered image layout.

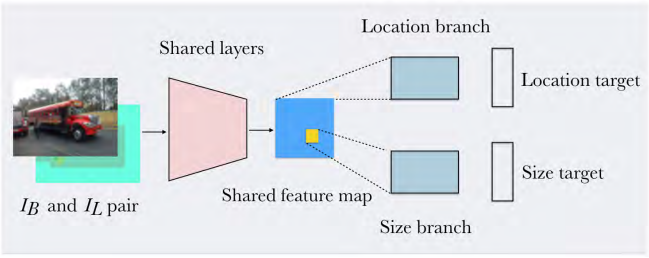


Fig. 4. Overview of the prediction model: the proposed network exploits a two branch architecture, with the first branch predicting the location and the second branch predicting the size.

palettes achieves similar performance. The layout image (indicated as I_L) represents the object layout of the image, as shown in Figure 3.

We additionally experimented with encoding the detected objects in a many-channel volume representation where each channel encodes a binary mask of all detected instances from each category. We find that the volume representation performs similarly with our representation based on a color palette encoding. We choose the simpler color palette representation due to its simplicity and compactness.

4.3 Prediction target

The target of our prediction model is the bounding box of a person instance. We first discuss how we represent the bounding box using normalized coordinates, and then explain how we discretize these coordinates for use in two classification tasks.

The bounding box representation from a ground truth annotation in the dataset is a four dimensional vector: (x_{min}, y_{min}, x_{max}, y_{max}), where (x_{min}, y_{min}) represents the top-left coordinate and (x_{max}, y_{max}) represents the bottom-right coordinate. For images of different resolutions, a normalized bounding box representation is required for consistent prediction. To do this, our system pads each rectangular image by the minimum amount so a square image is obtained, using a padding color that is the mean color for the ImageNet dataset [Russakovsky et al. 2015]. The bounding box is first shifted to account for the square padding, then transformed into normalized coordinates $(x_{\text{stand}}, y_{\text{stand}}, w, h) \in [0, 1]$, where $x_{\text{stand}} = \frac{1}{2s}(x_{\text{min}} + x_{\text{max}})$, $y_{\text{stand}} = \frac{1}{s} y_{\text{max}}$, s is the width of the square image, and w, h are the width and height of the box relative to the square image. Thus, $(x_{\text{stand}}, y_{\text{stand}})$ is the lowest center (standing) point of the bounding box.

Direct regression in a four dimensional continuous space is challenging. To facilitate the bounding box prediction, our system discretizes the (x, y) location domain into a 15×15 grid board, and then represents $(x_{\text{stand}}, y_{\text{stand}})$ as the index of the grid g_{xy} where it is located (concretely, g_{xy} is a class, which could be represented by an integer). Similarly, the (w, h) size domain is also discretized so that (w, h) is represented as another grid index g_{wh}. By doing this, we formulate the bounding box prediction as two classification problems with 225 (15 x 15) different classes for each.

4.4 Prediction model

Given I_B, I_L as inputs and g_{xy}, g_{wh} as targets, our next challenge is to learn the underlying mapping between them. Our approach is to learn the location (g_{xy}) and size (g_{wh}) simultaneously as they are highly correlated. In particular, we develop a novel CNN-based model which can be trained in an end-to-end manner.

In our model, the images (I_B, I_L) are first concatenated along the depth channel, and fed through a shared front-end network, as shown in Figure 4. This network is shared in the sense that the same weights are used before the split into the location and size branches. The shared network contains three residual bottleneck modules with projection shortcuts, similarly as in He et al. [2016]. These shortcut connections enable the model to leverage both low-level and high-level cues learned by the model. Starting from the output feature map of the shared network, the model is then separated into two smaller branches, with the first branch predicting the location (g_{xy}),

Layers	Activation Size
Input	6 x 480 x 480
64 x 7 x 7 conv, stride 2	64 x 237 x 237
3 x 3 maxpooling, stride 2	64 x 118 x 118
conv block, (64, 64, 128) filters + shortcut	128 x 59 x 59
conv block, (64, 64, 128) filters + shortcut	128 x 30 x 30
conv block, (128, 128, 512) filters + shortcut	512 x 15 x 15

(a) Shared layers

Layers	Activation Size
Shared features	512 x 15 x 15
64 x 3 x 3 conv, dilation 2	64 x 15 x 15
1 x 3 x 3 conv, dilation 2	15 x 15

(b) Location prediction branch

Layers	Activation Size
Shared features	512 x 15 x 15
512 x 3 x 3 conv, dilation 2	512 x 15 x 15
ROI slicing	512 x 3 x 3
Global maxpooling	512
Linear, 225 filters	225
Linear, 225 filters	225

(c) Size prediction branch

Table 1. Network architecture of the proposed prediction model. Here "shortcut" indicates the projection shortcut of He et al. [2016].

and the second branch predicting the size (g_{wh}). These two branches also incorporate dilated convolutional layers introduced in [Yu and Koltun 2016] in order to use larger receptive fields without using an additional number of parameters. Table 1 lists the layer-by-layer details of the proposed network architecture.

Note that the size of the predicted box should be consistent with the local context. For instance, person segments should not be larger than instances of larger objects appearing in their surroundings, such as buses or cars. Therefore, in the size prediction branch, after a small 3 x 3 dilated convolution, our system first remaps the normalized coordinates $(x_{\text{stand}}, y_{\text{stand}})$ into the spatial coordinates of the output activations, to obtain grid coordinates $(x_{\text{grid}}, y_{\text{grid}})$. A (3 x 3 x 512) activation slice is then extracted along the depth channel. This is done by extracting activations from a box with 3x3 spatial size, such that the lowest center coordinate of the box is $(x_{\text{grid}}, y_{\text{grid}})$. Here 512 is the number of filters in the activation map. We call this process as Region of Interest (ROI) slicing. This smaller activation map is then fed through the rest of the layers of the size branch. By doing this, the size prediction network attends to a sub-region of the feature map that captures the local context of the potential segment.

One subtle point for this design is that, during training, the normalized coordinates $(x_{\text{stand}}, y_{\text{stand}})$ we use for ROI slicing come from the ground truth bounding box. However, during testing, $(x_{\text{stand}}, y_{\text{stand}})$ are generated from the location we predict. Therefore, during inference our network is separated into two stages: the first stage just predicts the location of the segment; the second stage predicts the size based on the location predicted in the first stage.

4.5 Implementation

The proposed network is implemented in Keras [Chollet et al. 2015]. Most of the layers we exploit in our network (as shown in Table 1) have corresponding implementations in Keras except for the ROI slicing operation. The inputs of the network are the inpainted and blurred color images as well as their corresponding rendered layouts. These images are first transformed into square images by padding, as described before, and then resized to a resolution of (480, 480, 3). Before feeding the images into the network, the mean pixel color for the ImageNet dataset (in sRGB space, (103.939, 116.779, 123.68)) is subtracted from the image. To train the network, we use the Adam solver [Kingma and Ba 2015] with a fixed learning rate of 0.0001. We restrict the training data to the training split of MS-COCO dataset and use horizontal flipping for data augmentation. Our final model requires four epochs of training.

5 PERSON SEGMENT RETRIEVAL AND COMPOSITING.

In this section, we introduce a simple but efficient context based person segment retrieval and compositing scheme based on a hybrid deep feature representation. We first discuss how we create the pool of candidate person segments and perform the retrieval. Then we describe how we perform compositing.

5.1 Creating the candidate pool of person segments

To build a candidate pool for person segment retrieval, we use the annotated data from the validation split of the MS-COCO Dataset. We chose this split because these images are also held out from the training of the bounding box prediction. We apply the same filtering routines as in Section 4.1 to exclude segments that are heavily occluded, small or incomplete. Finally, we manually filter the remaining segments to remove partially occluded instances.

Although these segments come with ground truth segmentation annotations, most of the annotations are not accurate enough for compositing applications. Therefore, we also perform manual segmentation using the Magnetic Lasso Tool from Adobe Photoshop. We only need to do this once for each human segment. In total, our candidate pool contains 4000 person segments.

5.2 Context based person segment retrieval

Given a background image and a predicted bounding box, our goal is to retrieve a person segment from the candidate pool that not only matches the global scene semantics but also appears compatible with the local context. Various hand-crafted feature descriptors [Torralba et al. 2003][Hu and Collomosse 2013] have been proposed to facilitate image retrieval. Recently, using intermediate neural network activations as feature representations has shown to perform competitively for various semantic retrieval tasks even when the underlying network has been pre-trained in an unrelated classification task [Babenko et al. 2014]. However, previous methods mostly aim to retrieve images that "look similar" with respect to a query image, while our goal is to retrieve segments which are not present but "look natural" when composited on a background scene.

Fig. 5. Person segment retrieval: given the input image (A) and the predicted bounding box (the green box in (B)), the proposed system incorporates features from both the global scene and the local context (covered by the yellow box in (B)) to help retrieve a favorable person segment (within the yellow box in (C)) and composite it on the input image automatically.

Our key insight here is that, by incorporating the contextual information of both the query background image and the candidate person segments, we could adapt and extend feature-based methods to retrieve segments from images which share similar global scene semantics and local context with the background image. Specifically, for each input image, our system first extracts deep features which describe global scene semantics of the background image. The deep feature descriptor we adopt is the activation map from the mean pooling layer of ResNet50 [He et al. 2016], which results in a 2048-dimensional feature vector. Similarly, for each candidate person segment, we extract the same feature descriptor for its corresponding background image. Measuring the distance between the input image and the source images of the candidates in feature space can help retrieve segments appearing in similar scenes. However, the retrieved segment does not necessarily look natural in the local context if only global compatibility is considered.

To further enforce the local compatibility, given the predicted bounding box, our system crops a local image patch which shares the same center with the bounding box but is twice as large in both width and height, as shown in Figure 5. The same feature descriptor (activations of the mean pooling layer of ResNet50) of this local patch is then extracted. For each candidate person segment, our system extracts similar local feature descriptors. Measuring the distance between these local features can help retrieve segments appearing in similar local contexts as in the target location. As an additional benefit, when incorporating these local features, we observed that the retrieved segments also exhibit similar lighting conditions as in the input context. This makes the later processing stages of composition and relighting easier or in some cases unnecessary.

In our implementation, the segment retrieval proceeds in two steps: (1) our system first filters the segments whose bounding box sizes are quite different from the query box size. In order to do this, our system aligns the centers of the query and target bounding boxes and computes their Intersection over Union (IoU). Segments with IoUs smaller than 0.4 are excluded. (2) From the remaining candidate segments in our collection, the system retrieves the top one segment that is “closest" to the query input in feature space. Specifically, we use a cosine distance between the query input and the target segment, each represented by a concatenation of both the global and local feature descriptors. To accelerate the retrieval process, we also build a kd-tree structure of the candidate segments based on the proposed distance metric.

5.2.1 Feature selection. For this retrieval task, we experimented with a few different feature descriptors (e.g. GIST feature [Torralba et al. 2003], unsupervised learned feature from the Context Encoder work [Pathak et al. 2016], deep activation maps from VGG16 [Simonyan and Zisserman 2014] and ResNet50), we adopted the mean pooling feature from ResNet50 based on several observations: (1) compared with GIST feature and the context encoder feature [Pathak et al. 2016], the superiority of deep features was demonstrated by a pilot user study we conducted on Amazon Mechanical Turk. The setup of the pilot study was similar with the one we're going to introduce in Section 6.2. (2) different from VGG16 net, ResNet50 incorporated Batch Normalization layers [Ioffe and Szegedy 2015], which produce activation maps with similar magnitude scales for different dimensions. This is important when measuring the distance between feature descriptors. In fact, as the feature maps from VGG16 net exhibited various magnitude scales for different dimensions, our experiments showed that they usually resulted in poor retrieved segments. (3) compared with other layers in ResNet50, the activation map from the mean pooling layer encodes much semantic information (it is one layer before the final classifier) in smaller dimensions (2048), which makes it both effective and efficient for our retrieval task.

5.3 Compositing

With the retrieved segment in hand, our system scales and composites it onto the background such that the segment has the same center and height as the predicted bounding box. Although the segment already has a clean binary mask produced from the Photoshop magnetic lasso tool we discussed in Section 5.1, we apply an off-the-shelf alpha matting method [Chen et al. 2013] to obtain smooth natural transitions over the composite region. We also experimented with applying global relighting methods such as [Tsai et al. 2017; Xue et al. 2012]. However, as described previously, our retrieved segments usually exhibit similar lighting conditions with the background scenes, so we find that in general such global adjustments do not significantly improve the visual quality of the composite outputs. Figure 6 shows example composite results produced by our method covering various scenes.

6 EVALUATION

6.1 Quantitative Evaluation of Bounding Box Prediction

During training of the bounding box prediction model, we use the ground truth bounding boxes as the target for supervised learning. At first glance, it may seem reasonable to use evaluation metrics from object detection systems, such as average precision or precision-recall (PR) curve. However, for each specific background

Fig. 6. Composites automatically generated from our system. (A)(C): input images, (B)(D): composite results.

image, there may be multiple locations suitable for composing person instances with various sizes. The goal of the prediction model is to learn the distribution of feasible object layouts instead of overfitting toward the exact ground truth boxes in the dataset. In fact, we try to avoid this situation by blurring the input image so that the system can not overfit to inpainting artifacts.

Therefore, to evaluate the performance of the bounding box prediction model, we propose to measure the correlation between the distributions of the predicted boxes and the ground truth boxes. In particular, we represent the distribution of bounding boxes as two 2D histograms: A position histogram for the $(x_{\text{stand}}, y_{\text{stand}})$ coordinates, and a size histogram for the (w, h) sizes. The bin sizes we use for histogram computations are both 15×15, the same resolution we use for the prediction model.

For this experiment, the ground truth bounding boxes we use are from the validation split of the MS-COCO Dataset, which are held out from the training stage. The generated boxes are predicted from the same set of images but with the person segments erased and inpainted. To measure the correlation of the histograms, we use a correlation metric defined as:

$$d(H_g, H_p) = \frac{\sum_{i=1}^{N}(H_g(i) - \overline{H_g}) \cdot (H_p(i) - \overline{H_p})}{\sqrt{\sum_{i=1}^{N}(H_g(i) - \overline{H_g})^2 \cdot \sum_{i=1}^{N}(H_p(i) - \overline{H_p})^2}} \quad (1)$$

$$\bar{H_k} = \frac{1}{N}\sum_{j=1}^{N} H_k(j), \quad (2)$$

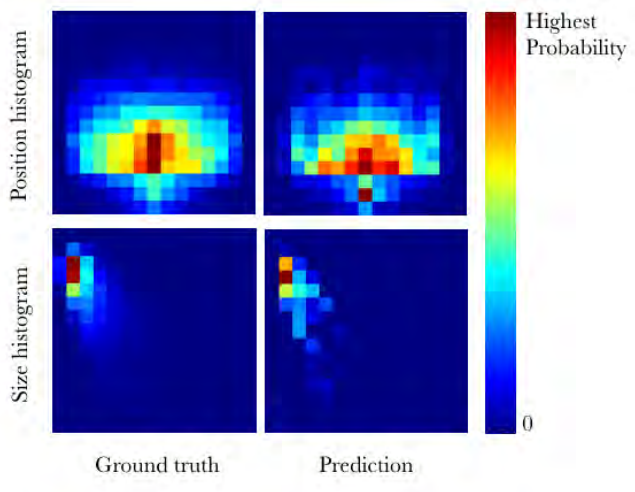


Fig. 7. Ground truth bounding box statistics (first column) and statistics measured from our prediction model (second column). When measuring the position distribution, the correlation between the ground truth and the prediction is 0.9458. When measuring the size distribution, the correlation between the ground truth and the prediction is 0.9378.

Model	Mean score	Mean time (s)
baseline-1	0.179±0.023	1.345
baseline-2	0.199±0.025	1.407
baseline-3	0.275±0.030	1.702
top-1	0.439±0.031	1.768
best of top-8	0.512±0.034	1.785
real	0.896±0.015	1.658

Table 2. Quantitative results from the user study.

where H_g and H_p represent the histograms of the ground truth and the predictions respectively, and N is the total number of bins, which is 225 in our case.

Under this proposed metric, the correlation between the ground truth and the prediction is 0.9458 when measuring the position histograms, and 0.9378 when measuring the size histograms. As judged by these high correlation scores, our prediction model can mimic reasonably well real person layouts in natural images. Figure 4 shows the 2D histograms we use for this evaluation.

6.2 Qualitative Evaluation via User Study

To evaluate the visual realism of the composite images, we conduct a human subject study using Amazon Mechanical Turk.

6.2.1 Models to be evaluated. For comparison purposes, three baseline methods are also evaluated, as described below.

(1) Baseline-1: the bounding box is sampled from the ground truth distribution, and the segment is retrieved using our proposed segment retrieval method. The purpose of this baseline is to evaluate the impact of our bounding box prediction method;
(2) Baseline-2: the bounding box is predicted by our system, then a segment is randomly sampled from the candidate pool. The purpose of this baseline is to evaluate the impact of our segment retrieval method;
(3) Baseline-3: the bounding box is predicted by our system but the segment is retrieved using a global GIST feature [Torralba et al. 2003] under Euclidean distance, resembling the work of [Hays and Efros 2007]. The purpose of this baseline is to evaluate the effect of deep feature representations on this problem.

For our method, we evaluate the top-1 composite from our system. We also include a manually chosen "best" images of the top-8 outputs based on the following criteria: combinations of the top-2 location predictions, top-2 size predictions and top-2 retrieved segments. For each background image we evaluate five composite images.

6.2.2 User study setup. During the study, the participants were presented with a sequence of images and instructed to press the R key if the user considered the image to be real or the F key if it was considered to be fake. For each image, the user had to respond in 10 seconds, otherwise the data was ignored. To avoid interference effects, we showed each participant examples from only one method. To incorporate quality control, we also included real and obviously fake composite images so that the ratio between real images and composite images is 1:1. Only the responses from participants who obtained at least 80% accuracy on both the real images and obviously fake images were collected. For each composite image, if a participant thought it was real, it was assigned a score of 1.0, otherwise it received a score of 0.0. We averaged the scores of each composite image over 25 distinct responses.

6.2.3 Quantitative results. Table 2 shows the mean scores and the mean times the participants spent on each image. Standard errors for the scores were computed by applying bootstrapping to the means. We notice that both the top-1 and "best" of top-8 composites outperform all baseline methods in terms of the mean scores. The "best" of top-8 composites perform slightly better than top-1, and the participants also spent more time on them. However, we could also notice that, there is still a "gap" between the "best" of top-8 composites and real images. One explanation is that our current system has not considered shadows and lighting conditions explicitly, shadows being particularly challenging.

For the mean scores, we also tested for significance using a two-sided Student's t-test. The Holm-Bonferroni method was adopted to control the familywise error rate at the significance level of 0.05. We observe that the p-values of the t-tests between the baseline methods and ours were all smaller than 0.0002, which shows that our method is statistically significantly better than the baselines. However, the p-value between the top-1 and "best" of top-8 composites was 0.113, which didn't pass the Holm-Bonferroni test with a p threshold of 0.0128. While we conjecture that the "best" of top-8 composites should be better than top-1, we cannot demonstrate this with significance in our current user study without collecting much more data.

Fig. 8. Examples of the comparison for different methods: (A) input images; (B) from baseline-1; (C) from baseline-2; (D) from baseline-3; (E) Top-1; (F) "Best" of top-8. Note that for the first example (first row), the top-1 composite is also the "best" of the top 8 composites.

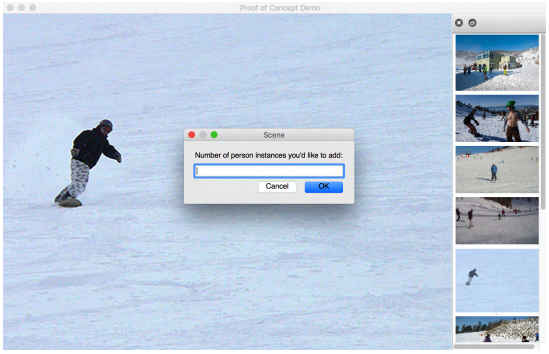


Fig. 9. A screen-shot of the proof-of-concept user interface.

Fig. 10. Potential application: composite image generation and interactive layout refinement. In our proof-of-concept user interface, given the input images in column (A), the users are prompted to determine how many person instances they would like to add to the background scene. The images in column (B) show the composite results from our automatic compositing system. The images in column (C) show the refinement results of (B) via user interactions. Please refer to the supplementary video for the whole generating pipelines of these examples.

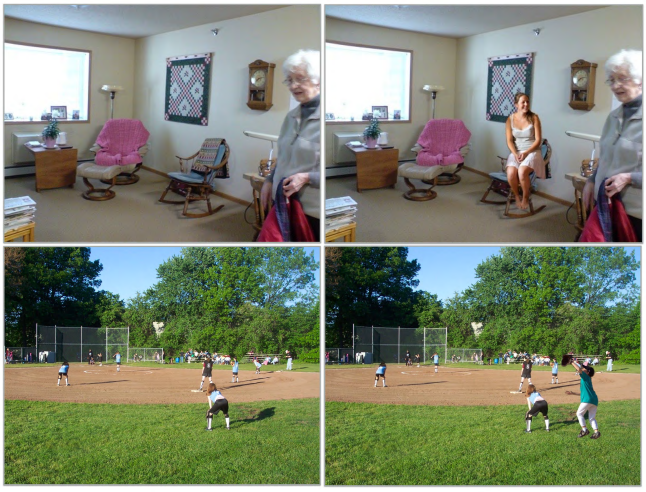
7 PROOF OF CONCEPT APPLICATION: COMPOSITE IMAGE GENERATION AND INTERACTIVE LAYOUT REFINEMENT

Using our system, we further develop a proof-of-concept user interface for composite image generation and interactive layout refinement. As discussed earlier, existing composite editing tools typically require intensive user interactions, which include providing the compatible foreground and background pairs, and finding the proper locations and sizes for composition. By incorporating the results from our automatic compositing system, our interface enables users to create and refine composite images with more automatic guidance and less manual searching for segments and proper positions and sizes.

In our current interface, given an input image, the user is prompted to determine how many person instances he or she would like to composite in the background scene. The top-1 automatic composite is then returned. For each of the predicted bounding boxes, 9 candidate segments are also cached in the program. The user can then refine the composite by replacing, translating or scaling each composited person segment in real-time. Figure 9 shows a screen-shot of the user interface. Figure 10 shows example results of automatic compositing and user refinement. Please refer to the supplementary video for screen recordings of the interactive editing.

8 LIMITATIONS AND FUTURE RESEARCH

Our current compositing system however effective, still has a few limitations. First, the MS-COCO dataset has certain biases toward some specific scenes, such as baseball field, ski slope, etc, thus limiting the appearance variation in our candidate pool of segments. This data bias problem makes it challenging to handle indoor scenes. Second, our bounding box prediction model depends on the performance of an external object detection system. There are situations



(A) (B)

Fig. 11. One of the limitations of our current system is that we have not modeled the interaction between the composited persons and their surroundings explicitly. In the first example (top-right), although our system retrieves a "sitting" person segment, it does not align well with the background chair. In the second example, the activity of the composited person is not compatible with the other person instances in the background. Here the first column are input images, the second column shows our composites.

where the results of the detection may hinder the predictions of our model. Third, while combining the global and local contexts helps retrieving segments that are compatible with the background, it does not guarantee that the retrieved segments "interact" correctly with each object instance in the scene. Figure 11 shows two examples when such interactions are important. In the first example, the retrieved "sitting" person does not align well with the chair. In the second example, the activity of the composited person is not compatible with the other person instances in the background. As a side effect of this limitation, when automatically compositing multiple person instances, it is possible that the interactions among the composited person segments are inconsistent. Finally, our system has not explicitly integrated lighting and shadow consistency with the background, which is another direction of future work.

9 CONCLUSION

We propose a fully automatic system for semantic-aware person composition. The proposed system accomplishes compositing by first predicting the bounding box of the potential instance and then retrieving a segment that appears compatible with the local context and global scene appearance. Quantitative and qualitative evaluations show that our system could predict person layouts for a given background scene and outperform robust baselines. We demonstrate the potential application of our system by developing a proof-of-concept user interface for interactive scene composition. For future research, there are still open challenges such as modeling the interaction between different object instances within the same scene, and handling a larger set of object categories beyond people.

REFERENCES

Artem Babenko, Anton Slesarev, Alexandr Chigorin, and Victor Lempitsky. 2014. Neural Codes for Image Retrieval. *European Conference on Computer Vision (ECCV)* (2014).

Moshe Bar and Shimon Ullman. 1996. Spatial Context in Recognition. *Perception* 25, 3 (1996), 343–352.

Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. 2009. PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing. *ACM Transactions on Graphics (SIGGRAPH)* 28, 3 (Aug. 2009).

Sean Bell, C. Lawrence Zitnick, Kavita Bala, and Ross Girshick. 2016. Inside-Outside Net: Detecting Objects in Context with Skip Pooling and Recurrent Neural Networks. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).

Qifeng Chen, Dingzeyu Li, and Chi-Keung Tang. 2013. KNN matting. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 35, 9 (Sept 2013), 2175–2188.

Tao Chen, Ming-Ming Cheng, Ping Tan, Ariel Shamir, and Shi-Min Hu. 2009. Sketch2Photo: Internet Image Montage. *ACM Transactions on Graphics (SIGGRAPH)* 28, 5 (Dec. 2009), 124:1–124:10.

François Chollet and others. 2015. Keras. https://github.com/fchollet/keras. (2015).

S. K. Divvala, D. Hoiem, J. H. Hays, A. A. Efros, and M. Hebert. 2009. An empirical study of context in object detection. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2009).

James Hays and Alexei A Efros. 2007. Scene Completion Using Millions of Photographs. *ACM Transactions on Graphics (SIGGRAPH)* 26, 3 (2007).

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).

Rui Hu and John Collomosse. 2013. A Performance Evaluation of Gradient Field HOG Descriptor for Sketch Based Image Retrieval. *Comput. Vis. Image Underst.* 117, 7 (July 2013), 790–806.

Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. 2017. Globally and Locally Consistent Image Completion. *ACM Transactions on Graphics (SIGGRAPH)* 36, 4, Article 107 (2017), 107:1–107:14 pages.

Sergey Ioffe and Christian Szegedy. 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *International Conference on International Conference on Machine Learning (ICML)*. 448–456.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations (ICLR)* (2015).

Jean-François Lalonde and Alexei A. Efros. 2007. Using color compatibility for assessing image realism. *IEEE International Conference on Computer Vision (ICCV)* (2007).

Jean-François Lalonde, Derek Hoiem, Alexei A. Efros, Carsten Rother, John Winn, and Antonio Criminisi. 2007. Photo Clip Art. *ACM Transactions on Graphics (SIGGRAPH)* 26, 3 (August 2007), 3.

T.Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. 2014. Microsoft COCO: Common Objects in Context. *European Conference on Computer Vision (ECCV)* (2014).

Tomasz Malisiewicz and Alexei A. Efros. 2009. Beyond Categories: The Visual Memex Model for Reasoning About Object Relationships. *Advances in Neural Information Processing Systems (NIPS)* (December 2009).

Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei Efros. 2016. Context Encoders: Feature Learning by Inpainting. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).

Patrick Pérez, Michel Gangnet, and Andrew Blake. 2003. Poisson Image Editing. *ACM Transactions on Graphics (SIGGRAPH)* (2003), 313–318.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *Advances in Neural Information Processing Systems (NIPS)* (2015), 91–99.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)* 115, 3 (2015), 211–252.

K. Simonyan and A. Zisserman. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR* abs/1409.1556 (2014).

Alvy Ray Smith and James F. Blinn. 1996. Blue Screen Matting. *ACM Transactions on Graphics (SIGGRAPH)* (1996), 259–268.

T. M. Strat and M. A. Fischler. 1991. Context-based vision: recognizing objects using information from both 2D and 3D imagery. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 13, 10 (1991), 1050–1065.

Jin Sun and David Jacobs. 2017. Seeing What Is Not There: Learning Context to Determine Where Objects Are Missing. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017).

Antonio Torralba. 2003. Contextual Priming for Object Detection. *International Journal of Computer Vision (IJCV)* 53, 2 (2003), 169–191.

A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin. 2003. Context-based vision system for place and object recognition. *IEEE International Conference on Computer Vision (ICCV)* (2003).

Y.-H. Tsai, X. Shen, Z. Lin, K. Sunkavalli, X. Lu, and M.-H. Yang. 2017. Deep Image Harmonization. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*

(2017).

Jianxiong Xiao, Krista A. Ehinger, James Hays, Antonio Torralba, and Aude Oliva. 2016. SUN Database: Exploring a Large Collection of Scene Categories. *International Journal of Computer Vision (IJCV)* 119, 1 (2016), 3–22.

Su Xue, Aseem Agarwala, Julie Dorsey, and Holly Rushmeier. 2012. Understanding and Improving the Realism of Image Composites. *ACM Transactions on Graphics (SIGGRAPH)* 31, 4 (2012), 84:1–84:10.

Chao Yang, Xin Lu, Zhe Lin, Eli Shechtman, Oliver Wang, and Hao Li. 2017. High-Resolution Image Inpainting using Multi-Scale Neural Patch Synthesis. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017).

Fisher Yu and Vladlen Koltun. 2016. Multi-Scale Context Aggregation by Dilated Convolutions. *International Conference on Learning Representations (ICLR)* (2016).

Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman, and Alexei A. Efros. 2015. Learning a Discriminative Model for the Perception of Realism in Composite Images. *IEEE International Conference on Computer Vision (ICCV)* (2015).