Fashion Landmark Detection and Category Classification for Robotics

Thomas Ziegler[1,2], Judith Butepage[2], Michael C. Welle[2], Anastasiia Varava[2], Tonci Novkovic[1] and Danica Kragic[2]

***Abstract*—Research on automated, image based identification of clothing categories and fashion landmarks has recently gained significant interest due to its potential impact on areas such as robotic clothing manipulation, automated clothes sorting and recycling, and online shopping. Several public and annotated fashion datasets have been created to facilitate research advances in this direction. In this work, we make the first step towards leveraging the data and techniques developed for fashion image analysis in vision-based robotic clothing manipulation tasks. We focus on techniques that can generalize from large-scale fashion datasets to less structured, small datasets collected in a robotic lab. Specifically, we propose training data augmentation methods such as elastic warping, and model adjustments such as rotation invariant convolutions to make the model generalize better. Our experiments demonstrate that our approach outperforms state-of-the art models with respect to clothing category classification and fashion landmark detection when tested on previously unseen datasets. Furthermore, we present experimental results on a new dataset composed of images where a robot holds different garments, collected in our lab.**

***Index Terms*—Vision for Robotics, Cloth/Garment Classification, Data augmentation, Generalizations with Convolutional Neural Networks**

I. Introduction

As the interest for fashion items increases in online shopping and e-commerce, the need for automated image analysis in the fashion industry is growing. This application area requires many tasks to be automatized, such as clothing category classification, fashion landmark detection, image retrieval and similarity based recommendations. Following the creation of large-scale fashion datasets [1, 2, 3], significant progress has been made in fashion image analysis. Deep learning based models have achieved significant performance gain in clothing category classification [2, 4, 5, 6, 7], item recommendation [2, 8, 9], and retrieval [2, 10].

However, for robotic clothing manipulation, the collection of large-scale datasets proves to be more difficult. Robotic clothing manipulation includes tasks such as clothing category classification [11, 12, 13, 14, 15] and tasks that require fashion landmark detection, such as grasp point detection [16, 17], folding [18, 19], sorting [20], unfolding [21, 22], and dressing [19]. Compared to retail applications, the vast majority of existing work on clothing category classification in robotics uses custom datasets for evaluation. These datasets are often limited in the number of images and contain only a small number of different categories.

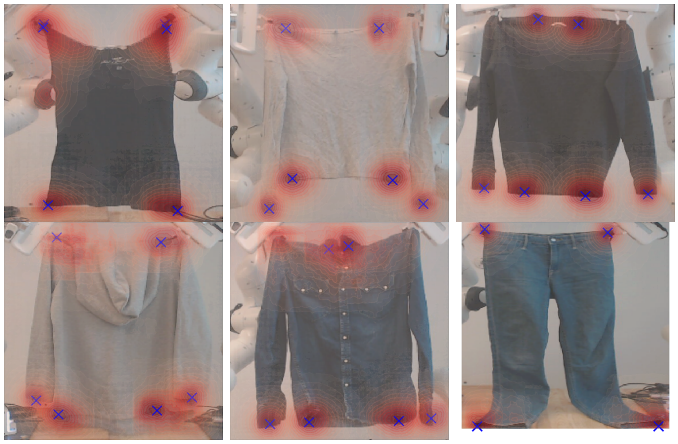


Fig. 1: Example images of the landmark localization on the six categories of our in-lab dataset. The categories are from top left to bottom right: Tank, Tee, Sweater, Hoody, Jacket, Jeans. Robot arms are visible in the images. The predicted heatmaps are shown in red and the blue crosses denote the selected maximum values.

In this work, we identify two tasks that are common to both retail and robotic applications, namely clothing category classification and fashion landmark detection. While the fashion industry often considers structured data, such as a human wearing clothes facing the camera, the data in robotic applications is less structured and can contain images of upside-down, crumpled clothing items. We build upon the progress made in fashion image analysis and propose a network architecture and training procedure on a large-scale fashion dataset DeepFashion [2]. Our model is capable to generalize well to the noisy, poorly controlled conditions encountered in robotic clothing manipulation. DeepFashion dataset contains over 280000 images of clothing separated into 46 categories and annotated with $4 \sim 8$ landmarks per item. In order to resemble the more challenging clothing configurations encountered in robotic manipulation, we introduce *elastic warping*, a novel image augmentation method. It uses random displacement fields to create authentic looking clothing configurations. Our proposed model incorporates rotation invariance and attention mechanisms in order to handle difficult configurations faced in robotic manipulation, such as random orientation.

[1]Thomas Ziegler and Tonci Novkovic, ETH Eidgenössische Technische Hochschule Zürich zieglert@ethz.ch, tonci.novkovic@mavt.ethz.ch

[2] Thomas Ziegler, Judith Butepage, Michael Welle, Anastasiia Varava, and Danica Kragic, KTH Royal Institute of technology butepage,mwelle,varava,dani@kth.se

The performance of our model is evaluated extensively on different publicly available datasets. One of these is a small-scale dataset created by us, which contains real world images typically encountered in a robotic manipulation task, *i.e*. where robot arms are visible in the image as shown in Figure 1. In contrast to other, state-of-the-art methods, our approach is able to generalize to new, much smaller datasets without additional fine-tuning. We illustrate a potential application scenario of our model by performing landmark detection on a garment that is being folded by a robot. We demonstrate that the landmarks are successfully detected even when the garment is partially occluded by the robot (see supplementary video[1]).

The landmark detection is very stable despite the robot occluding parts of the garment during the manipulation. Therefore we believe the proposed method is a first step for robotic clothing manipulation tasks that require basic visual information such as category classification and landmark detection.

The contributions of our work is fourfold:

(i) We propose a novel deep learning based network for clothing classification and landmark prediction specific for robot manipulation. To the best of our knowledge, this is the first deep learning based computer vision model for clothing manipulation trained solely on RGB images. *(ii)* We introduce *elastic warping* for landmark detection, a new data augmentation method that is capable of resembling more challenging clothing configurations which are not encountered in standard datasets. *(iii)* We provide extensive experimental results on different datasets which identify a lack of generalization to novel datasets of other state-of-the-art fashion networks. *(iv)* We created a small dataset, containing real world images of clothing in a realistic robotic manipulation environment. Additionally, we annotated landmark position in the CTU dataset [23]. The annotations, the in-lab dataset and the implementation are publicly available [1].

II. Related Work

We survey the work from the computer vision and robotic communities related to our work.

A. Fashion Networks

Fashion image analysis has drawn increased attention in the field of computer vision due to its impact on e-commerce and online shopping. Several deep learning networks evolved from this trend for the different analysis tasks of clothing recognition [2, 4, 5, 6, 7], recommendation [2, 8, 9], retrieval [10], and fashion landmark localization [2, 3, 7, 24, 25, 26]. In our work, the focus is on category classification and landmark localization.

Liu et al. [2] propose a multi-branch network for simultaneous classification, retrieval and landmark localization. In [3], the same authors propose a combination of three cascaded networks for a gradual refinement of landmark localization. Yan et al. [24] use recurrent spatial transformers in combination with selected dilated convolutions to predict landmarks in unconstrained scenes. More recently, Wang et al. [25] proposed a deep fashion grammar network for combined clothing category classification and landmark localization. The network encodes two attention mechanisms: landmark-aware and category-driven attention. A similar network was proposed by Liu and Lu [26] which has an increased resolution in the predicted heatmaps for the landmarks and uses a unified attention mechanism instead of two separate streams.

B. Computer vision for robotic clothing manipulation

Robotic clothing manipulation is a well established research area with pioneer work going back more than twenty years [27]. It can be seen as a pipeline of several consecutive steps to bring an item of clothing from an unknown state into a desired one (*e.g*. folded or sorted) [18]. A broad overview over methods used for visual grasp point localization, classification and state recognition is given in [28].

Compared to the structured data used for retail applications, the image data used in the robotics community is of a different nature. The items are either lying in a spread or crumpled state on a flat surface [11, 12, 20, 29] or they are in a hanging state when grasped by a robotic gripper [13, 15, 17, 30, 31, 32, 33].

A major difference to vision applications in the fashion industry is that the robotics community has mostly focused on task specific, handcrafted feature extraction, such as edges and corners [34] and wrinkles [35]. Additionally, due to the 3D nature of the manipulation task, the use of physics and volumental simulators is more common in robotics [14, 36].

Some recent methods [17, 31, 32, 33] use convolutional neural networks (CNN) instead of handcrafted features to classify hanging items of clothing. All models are shallow, containing a few convolutional layers followed by a few fully connected layers. Stria and Hlavác [15] use a CNN to create a global feature vector from depth maps. The CNN is trained on a large dataset with common 3D objects. For classification on a smaller clothing dataset, the CNN is used to extract features which are passed to a Support Vector Machine.

Our proposed network has a similar architecture as the networks proposed in [25, 26], but it has been extended to a more challenging clothing configurations encountered in robotic applications. To the best of our knowledge, this is the first deep learning based network designed as part of a robotic clothing manipulation pipeline that only uses RGB images.

III. Method

In this section we formulate the problem and introduce two augmentation methods to resemble clothing configurations encountered in robotic clothing classification and landmark localization tasks. Finally we give a detailed description of the proposed network.

A. Problem Formulation

Given an image $\mathbf{I}$ the goal is to simultaneously predict the landmark locations $\mathbf{L}$ and category classification $\mathbf{C}$. The landmarks are defined as $\mathbf{L} = \{(x_k, y_k)\}_{k=1}^{n_L}$, where (x_k, y_k) is the kth pixel coordinate position in $\mathbf{I}$ and n_L the total number of landmarks per image.

[1] https://github.com/ThomasZiegler/Fashion_Landmark_Detection_and_Category_Classification

The category classification $\mathbf{C} \in [0,1]^{n_C}$ satisfies $\sum_{i=1}^{n_C} C_i = 1$, where n_C is the number of categories depending on the used dataset.

B. Image augmentation

In order to make the available fashion datasets more representative for robotic applications we propose two types of data augmentation: image rotation and *elastic warping*. One challenge is to augment an image together with its landmarks. We define the image before transformation as input image $\mathbf{I}$ and the image after the transformation as transformed image $\tilde{\mathbf{I}}$. In both cases w, h stand for the width and height of the image respectively.

The transformation can be represented as a mapping of the pixels, $\forall(\tilde{x}, \tilde{y}) \in [1, w] \times [1, h]$:

$$\tilde{\mathbf{I}}(\tilde{x}, \tilde{y}) = \mathbf{I}\big(x(\tilde{x}, \tilde{y}), y(\tilde{x}, \tilde{y})\big). \tag{1}$$

Where x, y are the pixel location in the input image $\mathbf{I}$ and $\tilde{x}, \tilde{y}$ the pixel location in the transformed image $\tilde{\mathbf{I}}$. The clothing landmark locations $\mathbf{L} = \{(x_k, y_k)\}_{k=1}^{n_L}$ are a set of n_L specific pixel coordinates in the input image $\mathbf{I}$.

When $x(\tilde{x}, \tilde{y})$ and/or $y(\tilde{x}, \tilde{y})$ are non-integer, interpolation is needed. We apply the commonly used bilinear interpolation [37] in such a case.

1) Rotation: A rather simple but powerful augmentation is image rotation. Rotating images with a small angle is often used to increase the performance in classification and/or detection tasks [38]. When items of clothing lie on the ground or on a table, they can be in any orientation. We hence randomly sample an angle θ in the range $[0, 2\pi]$ for each rotation.

2) Elastic Warping: To resemble the distortion of loose items of clothing, we propose an elastic warping method. The method is similar to the elastic deformation proposed in [37] but is further extended to produce realistic, task-specific images and to allow for landmark detection.

The deformation is created by generating two random displacement fields $\mathbf{\Delta x}(\tilde{x}, \tilde{y})$ and $\mathbf{\Delta y}(\tilde{x}, \tilde{y})$. The whole augmentation is performed in four steps:

First: Sample n_S pixel positions uniformly in the transformed image: $\boldsymbol{\mathcal{S}} = \{(\tilde{x}_i, \tilde{y}_i)\}_{i=1}^{n_S}$.

Second: For each pixel location in $\forall(\tilde{x}_i, \tilde{y}_i) \in \boldsymbol{\mathcal{S}}$ sample a random displacement from a uniform distribution $\mathcal{U}(-\alpha, \alpha)$:

$$\mathbf{\Delta x}(\tilde{x}_i, \tilde{y}_i) \sim \mathcal{U}(-\alpha, \alpha),\ \mathbf{\Delta y}(\tilde{x}_i, \tilde{y}_i) \sim \mathcal{U}(-\alpha, \alpha). \tag{2}$$

All other entries in the displacement fields are set to 0.

Third: Convolve the two displacement fields with a Gaussian filter $\mathbf{G}$, $\forall(\tilde{x}, \tilde{y}) \in [1, w] \times [1, h]$:

$$\mathbf{\Delta \bar{x}}(\tilde{x}, \tilde{y}) = \mathbf{\Delta x}(\tilde{x}, \tilde{y}) * \mathbf{G}(\tilde{x}, \tilde{y}) \tag{3}$$

$$\mathbf{\Delta \bar{y}}(\tilde{x}, \tilde{y}) = \mathbf{\Delta y}(\tilde{x}, \tilde{y}) * \mathbf{G}(\tilde{x}, \tilde{y}) \tag{4}$$

where $*$ denotes the convolution operator. and $\mathbf{G}(\tilde{x}, \tilde{y})$ is a Gaussian filter with variance parameter σ.

Fourth: Use the smoothed displacement field to create the transformed image, $\forall(\tilde{x}, \tilde{y}) \in [1, w] \times [1, h]$:

$$\tilde{\mathbf{I}}(\tilde{x}, \tilde{y}) = \mathbf{I}\big(\underbrace{\tilde{x} + \mathbf{\Delta \bar{x}}(\tilde{x}, \tilde{y})}_{x(\tilde{x}, \tilde{y})}, \underbrace{\tilde{y} + \mathbf{\Delta \bar{y}}(\tilde{x}, \tilde{y})}_{y(\tilde{x}, \tilde{y})}\big) \tag{5}$$

Fig. 2: Example images of our proposed elastic warping with $n_S = 3$, $\alpha = 500$ and $\sigma = 40$. Top left is the original image, all others are transformed versions using different random seeds. Each landmark is marked with a red cross.

The strength of the distortion can be adjusted by the number of initially displaced pixels n_S, the scaling of the uniform distribution α and the smoothness of the Gaussian filter σ. We use $n_S = 3$, $\alpha = 500$ and $\sigma = 40$ in our experiments. Figure 2 shows some examples when using this configuration.

Landmark warping The displacement fields indicate where a pixel in the transformed image was located in the input image. Due to the random nature of these fields no inverse exists. That means that it is not trivial to know if/where the pixels of the input image are found in the transformed image. This is problematic for landmark warping, since their location is only defined for the input image. In the following we describe an efficient method for retrieving the landmark position in the transformed image.

For every landmark position $\mathbf{L}_k = (x_k, y_k)$ we find n possible pixels in the transformed image $\tilde{\mathbf{I}}$ which originated at or near the position of the landmark in the input image $\mathbf{I}$:

$$\boldsymbol{\mathcal{X}} = \underset{\forall(\tilde{x}, \tilde{y}) \in [1,w] \times [1,h]}{\text{argmin-}n} \ \text{sort}\, |\tilde{x} + \mathbf{\Delta \bar{x}}(\tilde{x}, \tilde{y}) - x_k| \tag{6}$$

$$\boldsymbol{\mathcal{Y}} = \underset{\forall(\tilde{x}, \tilde{y}) \in [1,w] \times [1,h]}{\text{argmin-}n} \ \text{sort}\, |\tilde{y} + \mathbf{\Delta \bar{y}}(\tilde{x}, \tilde{y}) - y_k|, \tag{7}$$

where argmin-n returns the n smallest values from a sorted set. Note that both $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{Y}}$ contain coordinate pairs $(\tilde{x}, \tilde{y})$. The value of n depends on the image size and the chosen parameters n_S, α, and σ in the elastic warping. We use $n = 200$ in our experiments. To get the transformed landmark $\tilde{\mathbf{L}}_k$ we need to find the coordinate pair $(\tilde{x}^*, \tilde{y}^*)$ that is either present in both $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{Y}}$ or the coordinate pair in $\boldsymbol{\mathcal{X}}$ with closest neighbor in $\boldsymbol{\mathcal{Y}}$.

We use the fact that the pixel coordinates are unique integer values and create a hash table for all coordinate pairs in one set. In the following, one can search for each pair in the other set if a key exist in the hash table, which reduces time complexity for existing coordinated pairs to $\mathcal{O}(n)$.

If the hash table does not return a valid value, no exact match exists in $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{Y}}$. In this case, one can create a kd-tree for all coordinate pairs in $\boldsymbol{\mathcal{Y}}$ and use kd-tree search [39]

to find the nearest neighbor for the coordinate pairs in $\boldsymbol{\mathcal{X}}$. The average search time for kd-tree search is $\mathcal{O}(n \log n)$.

C. Network Architecture

The main network architecture is loosely based on the VGG-16 [40] network structure similar to the networks proposed in [25, 26]. The structure can be seen in Figure 3a. Compared to the base VGG-16 network, several structural changes are included: rotation invariance layers (Section III-C1), a landmark localization branch (Section III-C2) and attention branches for classification (Section III-C3). As many components are inspired by prior work, we focus the discussion on the main components and direct interested readers to the Appendix for detailed network structure descriptions.

1) Rotation invariance: Orientation variation occurs more often in robotic clothing classification images than they do in fashion images. In order to account for this, we replace the 2D convolution in the *conv1* to *conv4* layers with Averaged Oriented Response Convolutions (A-ORConvs). They produce enriched feature maps with the orientation information explicitly encoded [41].

In our network (Figure 3b), we use the A-ORConvs with four orientation channels (*i.e.* $N = 4$). We use the same filter size and the same number of total channels when replacing the standard 2D convolution in the *conv1* to *conv4* layers. This means that the effective number of parameters of the A-ORConvs is only a quarter of the normal convolution blocks. In order to create rotation invariant features, a Squeeze-ORAlign (S-ORAlign) layer [41] is used to find the main response channel.

2) Landmark Localization Branch: The landmark localization branch is the same as proposed in [26]. The branch structure is depicted in Figure 3c.

The landmark localization branch can be trained separately from the classification. Given that the extracted feature map $\mathbf{F}$ is of dimension $w_f \times h_f \times n_L$, where w_f and h_f are width and height of the feature map and n_L is the number of landmarks, the ground-truth heatmap and the predicted heatmap for the kth landmark can be denoted by $\mathbf{M}_k \in [0,1]^{w_f \times h_f}$ and $\hat{\mathbf{M}}_k \in [0,1]^{w_f \times h_f}$ respectively. The landmark localization branch is trained using pixel-wise mean square differences,

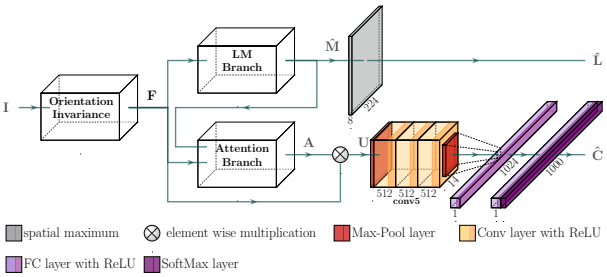
$$\mathcal{L}_{\text{LM}} = \sum_{i=1}^{n_B} \sum_{k=1}^{n_L} \sum_{x=1}^{w_f} \sum_{y=1}^{h_f} \|\mathbf{M}_k^i(x,y) - \hat{\mathbf{M}}_k^i(x,y)\|_2^2, \quad (8)$$

where n_B is the total number of training samples. The ground-truth heatmap $\mathbf{M}_k^i$ is generated by adding a 2D Gaussian filter at the corresponding location $\mathbf{L}_k^i$. Given a sample i the predicted coordinates for the kth landmark $\hat{\mathbf{L}}_k^i$ corresponds to the maximal value in the predicted heatmap,

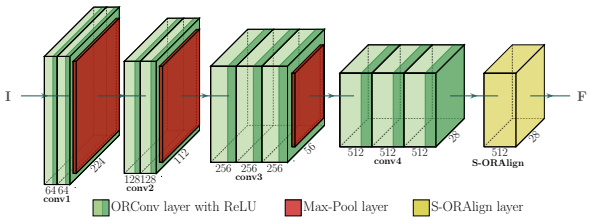
$$\hat{\mathbf{L}}_k^i \in \underset{(x,y)\in[1,w_f]\times[1,h_f]}{\text{argmax}} \hat{\mathbf{M}}_k^i(x,y). \quad (9)$$

If there is more than one maximum per landmark one of them is chosen at random.

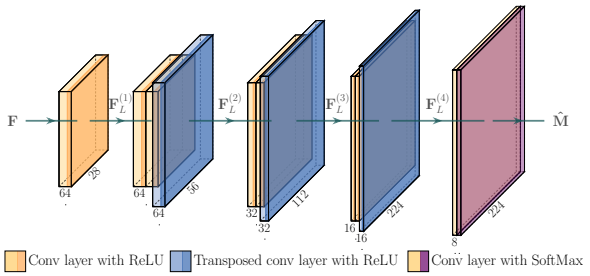
3) Attention Branch: The attention branch can be seen as a union of *spatial* attention [42] and *channel* attention [43]. The attention learns a saliency weight map $\mathbf{A}$. Inspired by



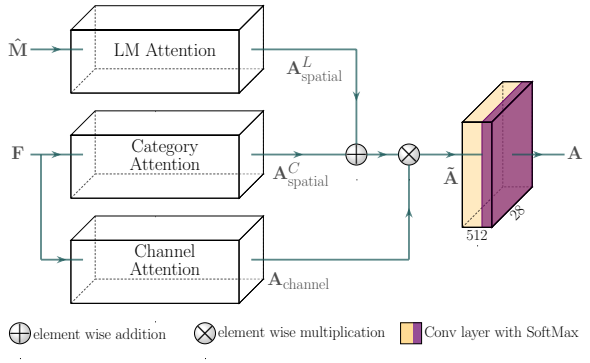
(a) Overall network structure



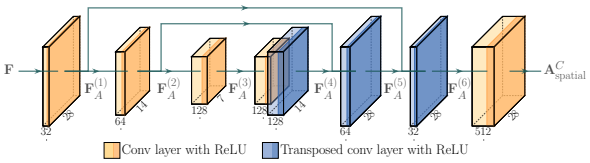
(b) Rotation invariance encoder



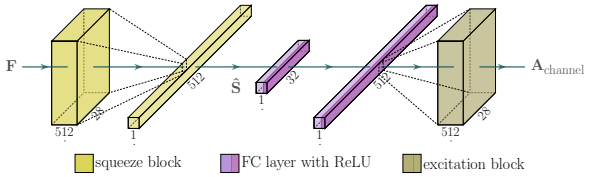
(c) Landmark (LM) localization branch



(d) Attention branch



(e) Category aware spatial attention



(f) Channel attention

Fig. 3: The different components of our model.

the proposed attention modules in [25] the spatial attention itself contains two types of attention, a landmark attention $\mathbf{A}^L_{\text{spatial}}$ and a category attention $\mathbf{A}^C_{\text{spatial}}$. Thus, the attention branch is designed as a three branch unit; two branches for the spatial attention $\mathbf{A}^L_{\text{spatial}}, \mathbf{A}^C_{\text{spatial}}$ (Figure 3d) and one for the channel attention $\mathbf{A}_{\text{channel}}$ (Figure 3f). These are combined in a factorized manner as explained below,

$$\mathbf{A} = (\mathbf{A}^L_{\text{spatial}} + \mathbf{A}^C_{\text{spatial}}) \times \mathbf{A}_{\text{channel}}. \tag{10}$$

Spatial Attention - Landmark Clothing landmarks represent functional regions of clothing and provide useful information about an item. The predicted heatmaps $\{\hat{\mathbf{M}}_k\}_{k=1}^{n_L}$ are used to guide attention to the functional clothing regions. The weight map is created by downsampling the predicted heatmaps which is followed by a max-pooling operation. This attention is learned in a supervised manner since it is directly derived from the predicted heatmaps.

Spatial Attention - Category Since the landmark attention only covers corner points of a clothing item, an additional spatial attention is used that focuses more on the clothing center. The category attention (Figure 3e) is modeled using an U-Net structure [44]. The model learns by itself which regions of an image are important. This is in contrast to our landmark attention, where the groundtruth heatmaps $\mathbf{M}$, which resemble the landmark attention, are provided during training.

Channel Attention The channel attention (Figure 3f) is implemented via a Squeeze-and-Excitation block [43]. First a *squeeze* operation creates $\mathbf{S}$, an embedding of the global distribution of the channel-wise feature responses. Then an *excitation* operation is performed on the channel wise aggregated feature map to create the channel attention. Following the proposal in [43] a bottleneck is created using two fully-connected layers, with a reduction rate r.

Factorization The factorization (Figure 3d) is performed by multiplying the channel-wise feature responses in the spatial attention with the corresponding channel weights, To refine the attention, an additional 1×1 convolution layer is added afterwards. This is motivated by the fact that the spatial and channel attention are not mutually exclusive but with co-occurring complementary relationship [45].

4) Output architecture: Given $\mathbf{A}$, we weight the S-ORAlign features $\mathbf{F}$, $\mathbf{U} = (\mathbf{1} + \mathbf{A}) \circ \mathbf{F}$, where $\circ$ denotes the Hadamard product and $\mathbf{1}$ is a tensor. Hence, features where $\mathbf{A}(\cdot,\cdot,\cdot) \in [-1, 0)$ are reduced and features where $\mathbf{A}(\cdot,\cdot,\cdot) \in (0, 1]$ are increased. Our attention incorporates semantic information and global information into the network helping to focus on important regions in the images. The features $\mathbf{U}$ are then fed in to the *conv5-1* layer. The rest of the network follows the VGG-16 structure.

IV. Experiments

This section describes several different experiments to evaluate the performance of the proposed network and learning procedure. The section starts with a description of the different datasets we use, followed by the descriptions of the individual experiments and results.

A. Datasets

In the following, we introduce all datasets used for training and/or evaluation.

1) DeepFashion dataset: The DeepFashion: Category and Attribute Prediction Benchmark (DeepFashion dataset[2]) [2] is a large collection of fashion images. It offers 289222 images collected from the Google image search engine and from shopping websites. The dataset contains 8 different landmarks (*i.e*. *left/right collar*, *left/right sleeve*, *left/right waistline*, and *left/right hem*), 46 clothing categories and 1000 clothing attributes. For each image a bounding box is provided. We use this dataset to train our network with our proposed augmentation methods and then perform inference on small-scale datasets.

2) CTU Color and Depth Image Dataset of Spread Garments: The CTU Color and Depth Image Dataset of Spread Garments (CTU dataset[3]) [23] is designed for testing and benchmarking garment segmentation and recognition. The dataset contains 1372 images of size 1280×1024 taken from a bird's eye perspective. There are 17 different items divided into 9 categories. Compared to the DeepFashion dataset the clothing items can be in any orientation and they contain not only flat spread but also wrinkled items. We manually labeled the landmark positions in each image. We use this dataset to train our network and evaluate its performance on more challenging clothing configurations typical in robotics. We also use it to evaluate the effect of our proposed augmentation methods when purely trained on the DeepFashion dataset.

3) In-Lab Dataset: As a first step towards generalizing classification and landmark detection results to images that are typical for robotic tasks, we created a small dataset. It contains 117 images from 6 different clothing categories (*i.e*. Tank, Tee, Sweater, Hoody, Jacket, Jeans). Each item is hold by two robotic arms at predefined grasping points (*i.e*. shoulders and waist). This state can be reached with an autonomous unfolding process as proposed in [21, 22]. The images are of size 960×720. Each item of clothing is captured in 9 different configurations of the robotic arms, such that they can overlap with the bounding box around the item. Furthermore, the background is not uniform and is partially cluttered. We annotated the images with the same landmarks as in the DeepFashion dataset and extracted a similar bounding box around each item. We use this dataset to evaluate the performance of our network on previously unseen items in a realistic lab environment.

B. Pretraining on the DeepFashion Dataset

In this section we describe the pre-training details for the DeepFashion Dataset. Experimental results on this dataset can be found in the Appendix.

1) Experimental Setup: We use the same settings as [2, 25, 26] for training and evaluation. In total 209222 images are used for training and 40000 images for validation. The final evaluation is performed on the remaining 40000 images. We

[2]http://mmlab.ie.cuhk.edu.hk/projects/DeepFashion/AttributePrediction.html

[3]https://github.com/CloPeMa/garment_dataset)

Methods (Trained on DF)	L.Collar	R.Collar	L.Sleeve	R.Sleeve	L.Waistline	R.Waistline	L.Hem	R.Hem	Avg.
Liu and Lu [26]	0.5056	**0.4810**	0.3288	0.2623	0.4908	**0.4665**	0.4047	0.4774	0.4272
Ours	**0.4972**	0.4835	**0.2846**	**0.2055**	**0.4870**	0.4677	0.4069	0.4727	**0.4131**
Liu and Lu [26] EW	0.5096	0.4995	0.3314	0.2626	0.4992	0.4730	**0.4063**	**0.4698**	0.4314
Ours EW	0.5194	0.5204	0.3538	0.2601	0.4935	0.5251	0.4185	0.4805	0.4464
Liu and Lu [26] R	0.0947	0.1004	0.0814	0.0670	0.1215	**0.1018**	0.2196	0.2177	0.1255
Ours R	0.1056	0.1075	0.0763	0.0708	0.1133	0.1206	0.1756	0.1526	0.1153
Liu and Lu [26] R & EW	**0.0863**	**0.0880**	0.0775	0.0717	**0.1030**	0.1265	0.2039	0.1860	0.1179
Ours R & EW	0.0999	0.0949	**0.0639**	**0.0581**	0.1039	0.1151	**0.1557**	**0.1474**	**0.1047**
Methods (Trained on CTU)	L.Collar	R.Collar	L.Sleeve	R.Sleeve	L.Waistline	R.Waistline	L.Hem	R.Hem	Avg.
Liu and Lu [26]	0.0560	0.0484	0.0473	0.0572	0.0473	0.0560	0.1010	0.0929	0.0632
Ours	0.0500	0.0801	0.0790	0.0745	0.0590	0.0713	0.0749	0.0853	0.0719
Liu and Lu [26] EW	0.0447	0.0442	0.0447	0.0481	0.0612	0.0826	0.0860	0.0780	0.0612
Ours EW	**0.0260**	**0.0267**	**0.0319**	**0.0262**	**0.0311**	**0.0359**	**0.0620**	**0.0548**	**0.0368**
Liu and Lu [26] R	0.0299	0.0314	0.0289	0.0335	0.0560	0.0402	0.0539	0.0460	0.0400
Ours R	**0.0181**	**0.0194**	**0.0253**	**0.0192**	0.0374	0.0382	**0.0314**	0.0383	0.0284
Liu and Lu [26] R & EW	0.0295	0.0277	0.0370	0.0403	0.0350	0.0561	0.0483	0.0509	0.0406
Ours R & EW	0.0199	0.0248	0.0348	0.0244	**0.0274**	**0.0204**	0.0334	**0.0276**	**0.0266**

TABLE I: Results on CTU dataset for landmark localization with different augmentation methods, when trained on the DeepFashion (DF) dataset (top) and in the CTU dataset (bottom). The values represent the normalized error (NE). Best results are marked in bold

Methods (Trained on DF)	Bluse		Hoody		Pants		Polo		Polo-Long		Skirt		Tshirt		Tshirt-Long		Overall	
	top-1	top-3	top-1	top-3	top-1	top-3	top-1	top-3	top-1	top-3	top-1	top-3	top-1	top-3	top-1	top-3	top-1	top-3
Liu and Lu [26]	35.00	50.00	31.58	52.63	33.33	52.28	33.33	52.28	33.33	52.28	36.84	52.63	33.33	52.28	33.33	52.28	33.74	52.15
Ours	20.00	50.00	21.05	47.37	19.05	47.62	19.05	47.62	19.05	47.62	21.05	52.63	19.05	47.62	19.05	47.62	19.63	48.47
Liu and Lu [26] R	45.00	70.00	42.11	68.42	42.86	71.43	42.86	71.43	42.86	71.28	42.11	68.42	42.86	71.43	42.86	71.43	42.94	70.77
Ours R	**85.00**	**90.00**	**84.21**	**89.47**	**85.71**	**90.48**	**85.71**	**90.48**	**85.71**	**90.48**	**84.21**	**89.47**	**85.71**	**90.48**	**85.71**	**90.48**	**85.28**	**90.18**
Liu and Lu [26] R & EW	55.00	75.00	52.63	73.68	57.14	76.19	57.14	76.19	57.14	76.19	52.63	73.68	57.14	76.19	57.14	76.19	55.83	76.46
Ours R & EW	80.00	**90.00**	73.68	**89.47**	76.19	**90.48**	76.19	**90.48**	76.19	**90.48**	78.95	**89.47**	76.19	**90.48**	76.19	**90.48**	76.69	**90.18**

TABLE II: Results on CTU dataset category classification with different augmentation methods, when trained on the DeepFashion (DF) dataset. Best results marked in bold.

use the normalized error (NE) [3] as the landmark localization error measure. This is the l_2 distance between the predicted and groundtruth landmark in normalized coordinates. For the category and attribute classification top-k classification accuracy is used.

The images are cropped using the provided bounding boxes. We train our model with and without our proposed data augmentation steps whereas the evaluation is always performed without augmentation. For implementation details, please see the Appendix.

C. Experiments on CTU Dataset

In this section we present experiments on the CTU Dataset. We perform two different types of experiments on the CTU dataset. In the first experiment, we analyze the inference performance of our network, solely trained on the entire DeepFashion dataset. Special interest lies in the proposed data augmentation methods, since the clothing configurations differ from the DeepFashion images. In the second experiment we evaluate the performance of our network when trained and evaluated on the CTU dataset.

1) Experimental Setup: It is important to note that the DeepFashion dataset has more than five times the number of categories than the CTU Dataset. Moreover the categories do not overlap exactly; if an item has a collar it is categorized as polo in the CTU dataset even though it might look more like a jacket than a polo shirt to a human. Furthermore, the CTU dataset distinguishes between long and short sleeve items, whereas DeepFashion does not (*e.g*. *tshirt* and *tshirt-long* can both be in the *Tee* category). We combine the categories as follows: bluse=(Blouse), hoody=(Hoodie, Sweater), pants=(Jeans, Jeggins, Joggers, Leggins), polo=(Tee, Button-Down), polo-long=(Button-Down, Henley, Jacket), skirt=(Skirt), tshirt=(Tee), tshirt-long=(Cardigan, Sweater, Tee). Since the DeepFashion dataset does not contain any towels, we ignore them in these experiments.

For the second experiment, we split the images randomly into a *train*, *validate*, and *test* set. (*i.e*. 787, 240, 270 images).

Both experiments are compared to the publicly available implementation of Liu and Lu [26]. For a fair comparison we train both models with the same augmentation methods (*i.e*. no augmentation, elastic warping (EW), rotation (R), and rotation & elastic warping (R & EW)). For implementation details, please see the Appendix.

2) Performance Evaluation: The results of landmark prediction and category classification on the CTU dataset with pre-trained models are shown in Table I (top) and II respectively. The benefit of training with rotated images becomes apparent. This is not surprising since the pictures of garments in the CTU dataset are taken in any possible orientation, whereas in the DeepFashion dataset all items of clothing are upright. Adding elastic warping increases the performance further for the landmark prediction for all cases except the one where training was performed on DeepFashion with no rotation. Additional results, presented in the appendix, show that adjusting the parameters of elastic warping can improve the performance further in some cases. Since it does achieve the best performance in the top-3 accuracy, we believe that a extended tuning of the elastic warping parameters could there-

Methods (Trained on DF)	L.Collar	R.Collar	L.Sleeve	R.Sleeve	L.Waistline	R.Waistline	L.Hem	R.Hem	Avg.
Liu and Lu [26]	0.0819	0.1061	**0.0910**	0.0975	0.0185	0.0175	0.0437	0.0788	0.0669
Ours	**0.0557**	**0.0682**	0.0947	0.1234	**0.0177**	**0.0135**	0.0497	0.0908	0.0642
Liu and Lu [26] EW	0.0910	0.1059	0.0915	**0.0470**	0.0341	0.0196	**0.0405**	0.0690	**0.0623**
Ours EW	0.0698	0.0923	0.1193	0.0843	0.0380	0.0315	0.0458	**0.0525**	0.0667
Liu and Lu [26] R	0.0620	0.0930	0.0924	0.0663	**0.0139**	0.0171	**0.0478**	0.1035	0.0620
Ours R	0.0621	**0.0767**	0.0949	0.0576	0.0527	**0.0134**	0.0926	0.0998	0.0687
Liu and Lu [26] R & EW	0.0657	0.1135	0.0892	**0.0523**	0.0163	0.0206	0.0586	**0.0662**	0.0603
Ours R & EW	**0.0532**	0.1129	**0.0827**	0.0535	0.0155	0.0202	0.0524	0.0817	**0.0590**

TABLE III: Results on in-lab dataset for landmark localization on unknown items of clothing. The values represent the normalized error (NE). Best result marked in bold.

Methods	Hoodie	Jacket	Sweater	Tank	Tee	Jeans	Overall
Liu and Lu [26]	00.00	**100.0**	84.21	**100.0**	55.56	**100.0**	71.65
Ours	05.88	88.89	**100.0**	**100.0**	62.96	96.30	76.07
Liu and Lu [26] R	00.00	11.11	**100.0**	77.78	62.96	**100.0**	66.67
Ours R	00.00	**100.0**	**100.0**	**100.0**	62.96	**100.0**	76.92
Liu and Lu [26] R & EW	00.00	77.78	**100.0**	**100.0**	**66.67**	**100.0**	76.07
Ours R & EW	**47.06**	88.89	94.74	**100.0**	51.85	96.30	**78.63**

TABLE IV: Classification accuracy on in-lab dataset for unknown items of clothing. Best result marked in bold.

fore increase the landmark prediction as well as classification performance. The overall classification accuracy of 85% shows that our model is able to generalize well even when trained on a dataset with significantly different configurations (*e.g.* items of clothing worn by persons) compared to 56% reached by Liu and Lu [26].

The results of the second experiment, trained and evaluated on CTU dataset, are shown in table I (bottom). Note that landmark predictions are significantly better when learned on the original dataset. The elastic warping seems to especially boost the performance in the case of no rotations. This is probably connected to the dataset composition and size as the EW augmented images boost the performance.

We omit the classification results since all the tested models achieve 100% accuracy.

Adding elastic warping as a data augmentation method can improve the performance in most of the evaluated cases. Our network outperforms the one proposed by Liu and Lu [26] when trained with the same augmentation methods in both experiments. This indicates that state-of-the-art methods are likely to not generalize well to more challenging datasets.

D. Experiments on In-Lab Dataset

In this experiment we analyze the inference performance of our network, solely trained on the DeepFashion dataset, on the images taken in a lab environment. For implementation details, please see the Appendix.

The results for landmark prediction and category classification are shown in Table III and IV respectively. Some landmark predictions are exemplified in Figure 1. There is one item (*i.e.* a hoody) that is in almost always misclassified, except when using elastic warping with our model. Furthermore, the long sleeve t-shirt (Figure 1 top row in the middle) is often classified as a sweater. With these two challenging items the best accuracy we achieve is 78.63%. Without these two items the accuracy increases to 93.33%. Due to the limited size of our dataset these two items have a significant impact. As the dataset is very limited in size, elastic warping can have a negative effect as well, as can be seen, for instance, in the drop in classification accuracy for the class Jacket. Further adjustments of the elastic warping parameters to achieve a higher similarity from the base images towards the task image might improve the results. Nevertheless, the combination of rotation and elastic warping leads to the best overall performance. The results of the landmark localization also show that our network is able to perform well even when an image contains parts of the robot. This can be seen in the video provided in the supplementary material[1], where a garment is being folded and the robotic arms occlude large parts of it.

Elastic warping improves the performance for Liu and Lu and leads to the best performance of our network in both the landmark localization and the classification. This indicates that the augmentation helps the models to generalize between the datasets. Furthermore, the model adjustments, such as rotation invariant convolutions, improve the state of the art methods [26] and make an important step towards there usage in robotic clothing manipulation.

V. Conclusion and Future Work

In this work we use a large publicly available fashion image dataset with data augmentation to train a network for garment classification and landmark detection for robotic manipulation application. This is the first work where a deep learning model trained on RGB images is used for clothing category classification and landmark detection for robotic applications. We show that our model is able to generalize to robotic specific item configurations which differ significantly from the training dataset. We achieve this by utilizing rotation and our newly proposed *elastic warping* augmentation method during training. After training, we use different datasets to evaluate the performance of our network. We observe that other state-of-the-art methods, while producing excellent results on the training set, are not able to generalize as well to novel datasets.

VI. Acknowledgements

This work has been supported by the Swedish Foundation for Strategic Research, Swedish Research Council and Knut and Alice Wallenberg Foundation.

VII. Appendix

In this appendix we give detailed account of the network structure and the experimental implementation details.

A. Network structure

We here describe the individual parts of the network in more detail.

1) Rotation invariance: In order to account for rotation, we replace the 2D convolution in the layers *conv1* to *conv4* of the VGG16 network with A-ORConvs layers. These produce enriched feature maps with the orientation information explicitly encoded [41]. A-ORConvs are an improvement of the Oriented Response Convolutions (ORConvs) initially proposed in [46]. These convolution blocks use Averaged Active Rotating Filters (A-ARFs) and Active Rotating Filters (ARFs), respectively. Both are a 5D tensors of size $n_O \times n_I \times w_f \times h_f \times N$, where n_O is the number of output channels, n_I the number of input channels, w_f and h_f are the width and height of the filter and N is the number of filter orientations. This means that in ARFs for each materialized filter, $N-1$ immaterialized rotated copies of the same filter are present. Therefore, during forward propagation one ARF produces a feature map of N channels with orientation information encoded. Depending on the orientation of the input image a different copy of the filter has the highest response. The improvement of A-ORConvs over ORConvs comes from reducing the risk of gradient explosion during training by updating the feature map with the mean value of the gradients from all its rotated copies instead of the sum of all gradients.

In our network, we use the A-ORConvs with four orientation channels (i.e. $N=4$). We use the same filter size and the same number of total channels when replacing the standard 2D convolution in the *conv1* to *conv4* layers. This means that the effective number of parameters of the A-ORConvs is only a quarter of the normal convolution blocks.

In order to get rotation invariant features S-ORAlign, proposed in [41], is used to find the main response channel. The S-ORAlign is inspired by the Squeeze-and-Excitation (SE) block [43], first a *squeeze* operation is performed by global average pooling. Then the main orientation channel is found via a maximum function and finally all channels are spun such that the main response channel is in the first position. The whole structure is depicted in Figure 4.

2) Landmark Localization Branch: The landmark localization branch is the same as proposed in [26]. The branch structure is depicted in Figure 6. It uses transposed convolutions [47] to produce heatmaps for all landmarks. The transposed convolutions allow for an upsampling of the S-ORAlign features $\mathbf{F} \in \mathbb{R}^{28\times28\times512}$ back to the original input image size. Given the features $\mathbf{F}$ a 1×1 convolution is applied to reduce the number of channels in the feature map to $\mathbf{F}_L^{(1)} \in \mathbb{R}^{28\times28\times64}$. Then three blocks of two 3×3 convolutions followed by a 4×4 transposed convolution are utilized. The padding and stride of the transposed convolution are 1 and 2, respectively. Hence, such a block upsamples the feature maps by a factor of two, at the same time the number of channels is reduced by a factor of two. Finally a 1×1 convolution with a sigmoid activation is used to convert the $\mathbf{F}_L^{(4)} \in \mathbb{R}^{224\times224\times16}$ feature map into the predicted heatmaps $\hat{\mathbf{M}} \in [0,1]^{224\times224\times8}$.

The landmark localization branch can be trained separately from the classification. Let $\mathbf{M}_k \in [0,1]^{224\times224}$ and $\hat{\mathbf{M}}_k \in [0,1]^{224\times224}$ denote the groundtruth heatmap and the predicted heatmap for the kth landmark, respectively. The landmark localization branch is trained using pixel-wise mean square differences,

$$\mathcal{L}_{\text{LM}} = \sum_{i=1}^{n_B}\sum_{k=1}^{8}\sum_{x=1}^{224}\sum_{y=1}^{224} \|\mathbf{M}_k^i(x,y) - \hat{\mathbf{M}}_k^i(x,y)\|_2^2, \tag{11}$$

where n_B is the total number of training samples. The groundtruth heatmap $\mathbf{M}_k^i$ is generated by adding a 2D Gaussian filter at the corresponding location $\mathbf{L}_k^i$. Given a sample i the predicted coordinates for the kth landmark $\hat{\mathbf{L}}_k^i$ corresponds to the maximal value in the predicted heatmap,

$$\hat{\mathbf{L}}_k^i \in \underset{(x,y)\in\{1,\ldots,224\}\times\{1,\ldots,224\}}{\text{argmax}} \hat{\mathbf{M}}_k^i(x,y). \tag{12}$$

If there is more than one maximum per landmark one of them is chosen at random.

3) Attention Branch: The attention branch can be seen as a union of *spatial* attention [42] and *channel* attention [43]. The attention learns a saliency weight map $\mathbf{A} \in [-1,1]^{28\times28\times512}$ of the same size as the S-ORAlign features $\mathbf{F} \in \mathbb{R}^{28\times28\times512}$. Inspired by the proposed attention modules in [25] the spatial attention itself contains two types of attention, a landmark attention $\mathbf{A}^L_{\text{spatial}}$ and a category attention $\mathbf{A}^C_{\text{spatial}}$.

We learn the spatial and channel attention in a factorized manner,

$$\mathbf{A} = (\mathbf{A}^L_{\text{spatial}} + \mathbf{A}^C_{\text{spatial}}) \times \mathbf{A}_{\text{channel}}. \tag{13}$$

The attention branch is designed in a three branch unit; two branches for the spatial attention $\mathbf{A}^L_{\text{spatial}}, \mathbf{A}^C_{\text{spatial}}$ (Figure 6a, Figure 6b) and one for the channel attention $\mathbf{A}_{\text{channel}}$ (Figure 6c). With the factorization (equation 13) combining them at the end, Figure 6d.

a) Spatial Attention - Landmark: Clothing landmarks represents functional regions of clothing and providing useful information about the item. The predicted heatmaps $\{\hat{\mathbf{M}}_k\}_{k=1}^8$ are used to get attention on the functional clothing regions. The weight map is created by downsampling the predicted heatmaps to the same size as the feature map in $\mathbf{F}$, followed by a max-pooling operation.

$$\hat{\mathbf{M}}' = \left\{\text{downsample}_8\ \hat{\mathbf{M}}_k\right\}_{k=1}^8 \tag{14}$$

$$\begin{aligned}\mathbf{A}^L_{\text{spatial}}(x,y) &= \max_k \hat{\mathbf{M}}'_k(x,y) \\ \forall(x,y) &\in \{1,\ldots,28\}\times\{1,\ldots,28\}\end{aligned} \tag{15}$$

This attention is learned in a supervised manner since it is directly derived from the predicted heatmaps.

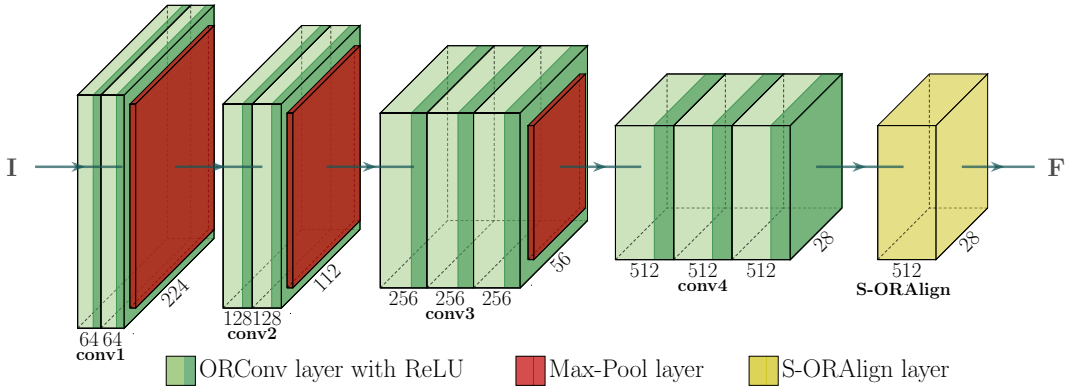


Fig. 4: Structure of the landmark localization branch. Each cuboid represents a feature map of the given layer. The number below a cuboid denotes the number of channels in the feature map and the number on the side denotes the width and height of the feature map. (Best viewed in color)

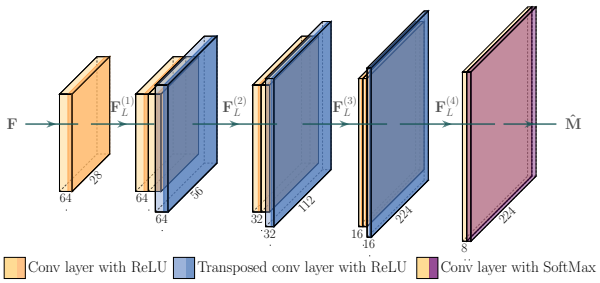


Fig. 5: Structure of the landmark localization branch. Each cuboid represents a feature map of the given layer. The number below a cuboid denotes the number of channels in the feature map and the number on the side denotes the width and height of the feature map. (Best viewed in color)

b) Spatial Attention - Category: Since the landmark attention only covers corner points of a clothing item, an additional spatial attention is used that focuses more on the clothing center. The category attention is modeled using an U-Net structure [44]. Given the S-ORAlign features $\mathbf{F} \in \mathbb{R}^{28\times28\times512}$ a 1×1 convolution is applied to convert the features into $\mathbf{F}_A^{(1)} \in \mathbb{R}^{28\times28\times32}$. The U-Net consists of a contracting path that consists of two 4×4 convolutions with stride 2, which squeeze the features down to $\mathbf{F}_A^{(3)} \in \mathbb{R}^{7\times7\times128}$. The number of feature channels doubles at every contracting step. Then a 1×1 convolution and 4×4 transposed convolution are applied generating the features $\mathbf{F}_A^{(4)} \in \mathbb{R}^{14\times14\times128}$. Followed by the U-Net expanding path, which consists of two 4×4 transposed convolution. The input of the transposed convolution is a concatenation of the output from the previous transposed convolution and the corresponding feature map from the contraction path. The number of feature channels halves at every expanding step. At the end a 1×1 convolution is used to convert the channels to the same number as in the S-ORAlign features.

The downpooling to a low resolution of 7×7 gives the spatial attention a large receptive field in the feature map of $\mathbf{F}$. The up-sampling is then used to have a weight map of the same size as $\mathbf{F}$. The model learns by itself which regions of an image are important. This is in contrast to our landmark attention, where the groundtruth heatmaps $\mathbf{M}$, which resemble the landmark attention, are provided during training.

c) Channel Attention: The channel attention is implemented via a Squeeze-and-Excitation block [43]. First a *squeeze* operation creates $\mathbf{S} \in \mathbb{R}^{512}$, an embedding of the global distribution of the channel-wise feature responses in $\mathbf{F}$. This channel descriptor is created using average pooling

$$S(c) = \frac{1}{28\times28}\sum_{u=1}^{28}\sum_{v=1}^{28}\mathbf{F}(u,v,c) \quad \forall c \in \{1,\dots,512\} \tag{16}$$

where $\mathbf{F}(\cdot,\cdot,c)$ is the feature map of the cth channel.

Then an *excitation* operation is performed on the channel wise aggregated feature map to create the channel attention.

$$\mathbf{A}_{\text{channel}} = \sigma\left(\mathbf{W}_2\,\text{ReLU}(\mathbf{W}_1\mathbf{S})\right), \tag{17}$$

where $\mathbf{W}_1 \in \mathbb{R}^{\frac{512}{r}\times512}$, $\mathbf{W}_2 \in \mathbb{R}^{512\times\frac{512}{r}}$, and σ represents the *sigmoid* activation function. Following the proposal in [43] a bottleneck is created using two fully-connected layers, with a reduction rate r. We choose $r = 16$ in all our experiments.

d) Factorization: The factorization is then performed by multiplying the channel-wise feature responses in the spatial attention with the corresponding channel weights,

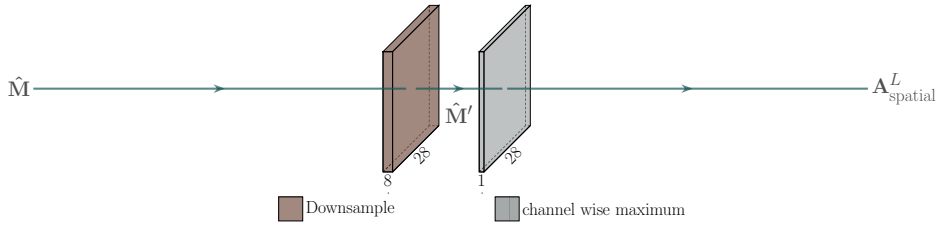
$$\tilde{\mathbf{A}}(x,y,c) = \left(\mathbf{A}^L_{\text{spatial}}(x,y,c) + \mathbf{A}^C_{\text{spatial}}(x,y,c)\right)\mathbf{A}_{\text{channel}}(c)$$
$$\forall(x,y) \in \{1,\dots,28\}\times\{1,\dots,28\} \quad \forall c \in \{1,\dots,512\}. \tag{18}$$

To refine the attention, an additional 1×1 convolution layer is added afterwards. This is motivated by the fact that the spatial and channel attention are not mutually exclusive but with co-occurring complementary relationship [45].

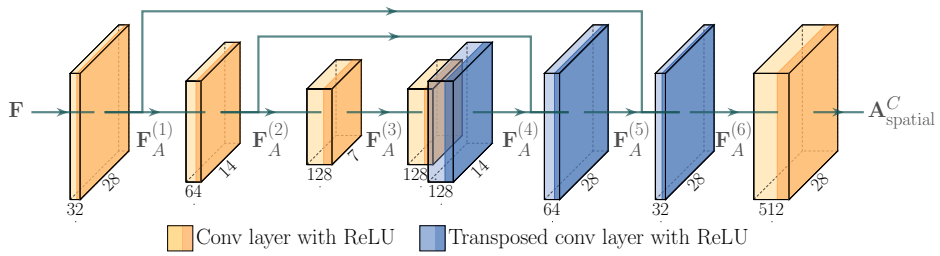
Afterwards, a *tanh* function is used to shrink the attention values into a range of $\mathbf{A} \in [-1,1]^{28\times28\times512}$.

4) Rest of the network: Given $\mathbf{A} \in [-1,1]^{28\times28\times512}$ we weight the S-ORAlign features $\mathbf{F} \in \mathbb{R}^{28\times28\times512}$,

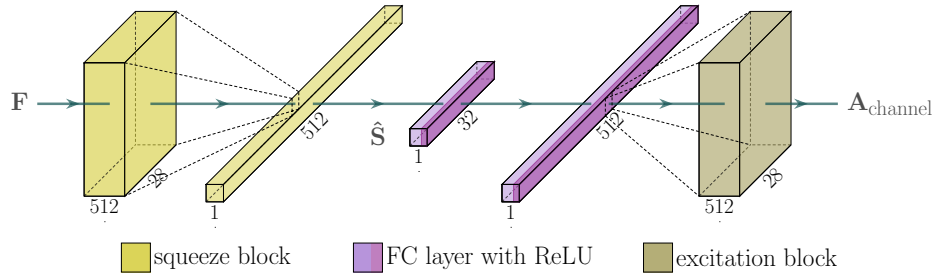
$$\mathbf{U} = (\mathbf{1} + \mathbf{A}) \circ \mathbf{F}, \tag{19}$$



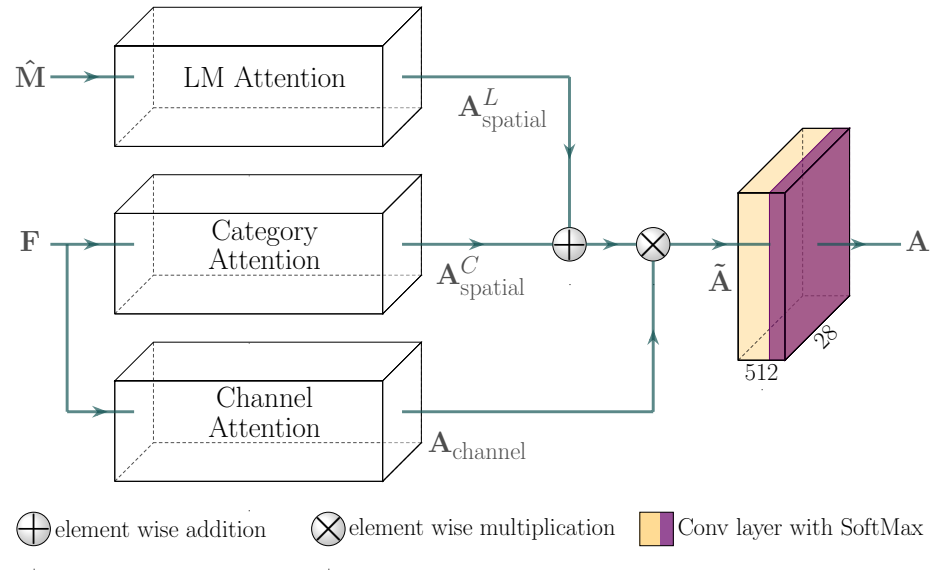
(a) Landmark aware spatial attention



(b) Category aware spatial attention



(c) Channel attention



(d) Overall attention branch

Fig. 6: The different components of the attention branch. The number below a cuboid denotes the number of channels in the feature map and the number on the side denotes the width and height of the feature map. (Best viewed in color)

where $\circ$ denotes the Hadamard product and $\mathbf{1}$ is a tensor of ones with size $28 \times 28 \times 512$. Hence, features where $\mathbf{A}(\cdot, \cdot, \cdot) \in [-1, 0)$ are reduced and features where $\mathbf{A}(\cdot, \cdot, \cdot) \in (0, 1]$ are increased. Our attention incorporates semantic information and global information into the network helping to focus on important regions in the images. The features $\mathbf{U}$ are then fed in to the *conv5-1* layer. The rest of the network follows the VGG-16 structure but with a reduced number of weights in the two fully connected layers (i.e. 1024 and 1000 instead of 4096 and 1000).

Method	best epoch
Ours	3
Ours rot.	47
Ours rot. & el. warp.	50

TABLE V: Number of epochs until early stopping (*i.e*. best result on validation set).

B. Experiments

In this section we describe the implementation details for each of the experiments and present results for the DeepFasion dataset.

1) DeepFashion Experiments:

a) Implementation Details: We build our network using the publicly available implementation[4] of Liu and Lu[2] as a starting point. The cropped images are resized to the input size of the VGG-16 network (*i.e*. 224×224). The A-ORConvs and normal convolution layers are pretrained on ImageNet [48]. The fully connected layers of the VGG-16 network are replaced with two separate fully connected layer branches, one for the category classification and the other for the attribute prediction. We use cross entropy loss for the category classification. Due to the inbalance between positive and negative samples asymmetric weighted cross entropy loss is used for the attribute prediction. The batch size is 32 and 64 during training and validation, respectively. The model is trained using the Adam optimizer [49] with an initial learning rate of 0.0002, which is multiplied by a factor of 0.8 every fifth epoch. The landmark detection branch is initially trained separately for 20 epochs. The landmark prediction is then locked and the learning rate is reset. Without locking the landmark prediction accuracy would decrease significantly during the classification training. The category classification and attribute prediction are trained for up to 50 epochs. We perform early stopping on the validation set. Meaning we track the best result on the validation set and stop the training if the result does not improve over 5 consecutive epochs. The model state that achieved the best result is then used in the evaluation on the *test* set. We do not perform specific parameter tuning depending on the dataset and/or the augmentation method. Furthermore, Table V shows the actually best epoch tracked with our early stopping.

b) Experimental results on DeepFashion: We compare our landmark prediction results to the following five models [2, 3, 24, 25, 26] and the clothing category classification to these models [2, 4, 5, 6, 25, 26, 50]. We copy the results in Table VI and X as they were presented in [26] and add our own results. We also show the results when using our proposed data augmentation methods during training.

One can see that we outperform all other system in the landmark localization task when no augmentation or rotation is used. This indicates that our rotation invariance network structure is generally beneficial. That is specially noticeable for *left/right sleeves*. These are the parts of clothing that generally have the most variation between images. On the other hand the category classification is not as good compared to the other systems. We assume that increasing the number of channels in the A-ORConv layers could increase the accuracy. This is because the actual number of feature channels is only a fourth due to the rotated copies. As we show in the experiments in the main paper, our pretrained network outperforms Liu and Lu [26] when tested on other datasets. This suggests that the state-of-the-art models are not able to generalize from the training dataset.

One can also see that our introduced *elastic warping* performs worse on this dataset. When trained on augmented data, the network spreads its computational power over more possible clothing configurations which might decrease performance on a certain configuration (the untransformed testing data).

2) Elastic Warping parameters Experiments: We run additional experiments for Lanmark detection trained on DeepFashion and CTU dataset evaluated on the CTU dataset. Table VII shows the result for $\alpha = 150$ and $\sigma = 10$, table VIII shows the result for $\alpha = 100$ and $\sigma = 10$, and IX shows the result for $\alpha = 200$ and $\sigma = 10$. We can clearly see that the EW helps to boost the performance for the R & EW augmentation when trained on the DeepFashion net and for only EW and R & EW for the CTU case. The best performance for when trained on the DeepFashion or CTU is achived with $\alpha = 100$ and $\sigma = 10$.

C. Implementation details for the CTU experiments

In the first experiment, we use the network trained as described in Section VII-B1 and perform solely inference with it. We do only consider the 13 categories in Table XI as possible predictions and mask the others out.

The network setup for the second experiment is the same as for the DeepFashion dataset described in Section VII-B1 with the exception of the following changes. The last fully connected layer of the VGG-16 network is reduced from 1000 categories down to 9 categories. The number of epochs is increased to a maximum of 200 since the dataset is much smaller and the learning rate decreases every 25th epoch. The landmark prediction is initially trained for 50 epochs.

1) Implementation details for the In-Lab Dataset: In this experiment we use the network trained as described in Section VII-B1. We perform solely inference with the network. For the evaluation we only consider the categories that are present and mask the rest out.

[4]https://github.com/fdjingyuan/Deep-Fashion-Analysis-ECCV2018

REFERENCES

[1] X. Liang, S. Liu, X. Shen, J. Yang, L. Liu, J. Dong, L. Lin, and S. Yan, "Deep Human Parsing with Active

Methods	L.Collar	R.Collar	L.Sleeve	R.Sleeve	L.Waistline	R.Waistline	L.Hem	R.Hem	Avg.
FashionNet [2]	0.0854	0.0902	0.0973	0.0935	0.0854	0.0845	0.0812	0.0823	0.0872
DFA [3]	0.0628	0.0637	0.0658	0.0621	0.0726	0.0702	0.0658	0.0663	0.0660
DLAN [24]	0.0570	0.0611	0.0672	0.0647	0.0703	0.0694	0.0624	0.0627	0.0643
Wang et al. [25]	0.0415	0.0404	0.0496	**0.0449**	0.0502	0.0523	**0.0537**	**0.0511**	0.0484
Liu and Lu [26]	**0.0332**	**0.0346**	0.0487	0.0519	**0.0422**	**0.0429**	0.0620	0.0639	0.0474
Ours	0.0343	0.0348	0.0488	0.0509	0.0436	0.0445	0.0582	0.0608	0.0470
Ours R	0.0351	0.0354	**0.0480**	0.0491	0.0440	0.0448	0.0564	0.0589	**0.0466**
Ours R & EW	0.0368	0.0383	0.0506	0.0517	0.0499	0.0524	0.0578	0.0610	0.0498

TABLE VI: Results on DeepFashion dataset for landmark localization. The values represent the normalized error (NE). Best results are marked in bold

Methods (Trained on DF)	L.Collar	R.Collar	L.Sleeve	R.Sleeve	L.Waistline	R.Waistline	L.Hem	R.Hem	Avg.
Liu and Lu [26]	0.5056	**0.4810**	0.3288	0.2623	0.4908	0.4665	0.4047	0.4774	0.4272
Ours	**0.4972**	0.4835	**0.2846**	**0.2055**	0.4870	0.4677	0.4069	0.4727	**0.4131**
Liu and Lu [26] EW	0.5123	0.5039	0.3440	0.2644	**0.4749**	0.5010	**0.4018**	**0.4660**	0.4335
Ours EW	0.5048	0.4982	0.3116	0.2893	0.4796	**0.4386**	0.4190	0.4708	0.4265
Liu and Lu [26] R	**0.0947**	0.1004	0.0814	**0.0670**	0.1215	**0.1018**	0.2196	0.2177	0.1255
Ours R	0.1056	0.1075	0.0763	0.0708	0.1133	0.1206	0.1756	**0.1526**	0.1153
Liu and Lu [26] R & EW	0.1077	0.1017	0.0873	0.0743	0.1215	0.1298	0.2187	0.2225	0.1329
Ours R & EW	0.1075	**0.0970**	**0.0718**	0.0715	**0.0976**	0.1083	**0.1505**	0.1569	**0.1076**
Methods (Trained on CTU)	L.Collar	R.Collar	L.Sleeve	R.Sleeve	L.Waistline	R.Waistline	L.Hem	R.Hem	Avg.
Liu and Lu [26]	0.0560	0.0484	0.0473	0.0572	0.0473	0.0560	0.1010	0.0929	0.0632
Ours	0.0500	0.0801	0.0790	0.0745	0.0590	0.0713	0.0749	0.0853	0.0719
Liu and Lu [26] EW	0.0395	0.0388	0.0448	0.0750	0.0452	0.0467	0.1064	0.0848	0.0602
Ours EW	**0.0263**	**0.0336**	**0.0273**	**0.0361**	**0.0431**	**0.0407**	**0.0483**	**0.0512**	**0.0383**
Liu and Lu [26] R	0.0299	0.0314	0.0289	0.0335	0.0560	0.0402	0.0539	0.0460	0.0400
Ours R	**0.0181**	**0.0194**	0.0253	**0.0192**	0.0374	0.0382	0.0314	0.0383	0.0284
Liu and Lu [26] R & EW	0.0319	0.0314	0.0332	0.0443	0.0559	0.0494	0.0442	0.0620	0.0440
Ours R & EW	0.0282	0.0251	**0.0230**	0.0291	**0.0179**	**0.0256**	**0.0293**	**0.0285**	**0.0258**

TABLE VII: Results on CTU dataset for landmark localization with different augmentation methods, when trained on the DeepFashion (DF) dataset (top) and in the CTU dataset (bottom). The values represent the normalized error (NE). Best results are marked in bold EW parameters: $\alpha = 150$, $\sigma = 10$.

Methods (Trained on DF)	L.Collar	R.Collar	L.Sleeve	R.Sleeve	L.Waistline	R.Waistline	L.Hem	R.Hem	Avg.
Liu and Lu [26]	0.5056	**0.4810**	0.3288	0.2623	0.4908	0.4665	0.4047	0.4774	0.4272
Ours	**0.4972**	0.4835	**0.2846**	**0.2055**	0.4870	0.4677	0.4069	0.4727	**0.4131**
Liu and Lu [26] EW	0.5014	0.5027	0.3287	0.2938	0.4997	0.4837	**0.3970**	0.4684	0.4344
Ours EW	0.5003	0.4829	0.2915	0.2334	**0.4689**	**0.4629**	0.4146	**0.4638**	0.4148
Liu and Lu [26] R	**0.0947**	0.1004	0.0814	0.0670	0.1215	0.1018	0.2196	0.2177	0.1255
Ours R	0.1056	0.1075	0.0763	0.0708	0.1133	0.1206	0.1756	**0.1526**	0.1153
Liu and Lu [26] R & EW	0.0961	0.0986	0.0830	0.0672	0.1082	0.1011	0.2161	0.2054	0.1220
Ours R & EW	0.0981	**0.0904**	**0.0689**	**0.0618**	**0.0838**	**0.0963**	**0.1530**	0.1643	**0.1021**
Methods (Trained on CTU)	L.Collar	R.Collar	L.Sleeve	R.Sleeve	L.Waistline	R.Waistline	L.Hem	R.Hem	Avg.
Liu and Lu [26]	0.0560	0.0484	0.0473	0.0572	0.0473	0.0560	0.1010	0.0929	0.0632
Ours	0.0500	0.0801	0.0790	0.0745	0.0590	0.0713	0.0749	0.0853	0.0719
Liu and Lu [26] EW	0.0395	0.0388	0.0448	0.0750	0.0452	0.0467	0.1064	0.0848	0.0602
Ours EW	**0.0261**	**0.0252**	**0.0264**	**0.0268**	**0.0330**	**0.0444**	**0.0536**	**0.0480**	**0.0354**
Liu and Lu [26] R	0.0299	0.0314	0.0289	0.0335	0.0560	0.0402	0.0539	0.0460	0.0400
Ours R	**0.0181**	**0.0194**	**0.0253**	**0.0192**	0.0374	0.0382	0.0314	0.0383	0.0284
Liu and Lu [26] R & EW	0.0214	0.0246	0.0300	0.0285	0.0412	0.0376	0.0439	0.0485	0.0345
Ours R & EW	0.0216	0.0186	0.0275	0.0237	**0.0252**	**0.0314**	**0.0239**	**0.0275**	**0.0249**

TABLE VIII: Results on CTU dataset for landmark localization with different augmentation methods, when trained on the DeepFashion (DF) dataset (top) and in the CTU dataset (bottom). The values represent the normalized error (NE). Best results are marked in bold EW parameters: $\alpha = 100$, $\sigma = 10$.

Template Regression," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 12, pp. 2402–2414, 2015.

[2] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, "DeepFashion: Powering Robust Clothes Recognition and Retrieval with Rich Annotations," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 1096–1104.

[3] Z. Liu, S. Yan, P. Luo, X. Wang, and X. Tang, "Fashion Landmark Detection in the Wild," in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp.

Methods (Trained on DF)	L.Collar	R.Collar	L.Sleeve	R.Sleeve	L.Waistline	R.Waistline	L.Hem	R.Hem	Avg.
Liu and Lu [26]	0.5056	**0.4810**	0.3288	0.2623	0.4908	0.4665	0.4047	0.4774	0.4272
Ours	**0.4972**	0.4835	0.2846	**0.2055**	**0.4870**	0.4677	0.4069	0.4727	**0.4131**
Liu and Lu [26] EW	0.5098	0.4999	0.3227	0.2810	0.5012	0.4949	**0.3944**	**0.4563**	0.4325
Ours EW	0.5174	0.4926	**0.2813**	0.2372	0.4918	**0.4393**	0.4366	0.4766	0.4216
Liu and Lu [26] R	**0.0947**	0.1004	0.0814	0.0670	0.1215	**0.1018**	0.2196	0.2177	0.1255
Ours R	0.1056	0.1075	**0.0763**	0.0708	0.1133	0.1206	0.1756	**0.1526**	0.1153
Liu and Lu [26] R & EW	0.1008	**0.0901**	0.0849	**0.0637**	0.1205	0.1134	0.2144	0.2096	0.1247
Ours R & EW	0.0977	0.1058	0.0801	0.0643	**0.0920**	0.1192	**0.1683**	0.1747	**0.1128**
Methods (Trained on CTU)	L.Collar	R.Collar	L.Sleeve	R.Sleeve	L.Waistline	R.Waistline	L.Hem	R.Hem	Avg.
Liu and Lu [26]	0.0560	0.0484	0.0473	0.0572	0.0473	0.0560	0.1010	0.0929	0.0632
Ours	0.0500	0.0801	0.0790	0.0745	0.0590	0.0713	0.0749	0.0853	0.0719
Liu and Lu [26] EW	0.0374	0.0404	0.0493	0.0627	0.0514	0.0390	0.0828	0.0818	0.0556
Ours EW	**0.0324**	**0.0323**	**0.0304**	**0.0472**	**0.0210**	**0.0179**	**0.0555**	**0.0515**	**0.0360**
Liu and Lu [26] R	0.0299	0.0314	0.0289	0.0335	0.0560	0.0402	0.0539	0.0460	0.0400
Ours R	**0.0181**	**0.0194**	0.0253	0.0192	0.0374	0.0382	0.0314	0.0383	0.0284
Liu and Lu [26] R & EW	0.0241	0.0270	0.0268	0.0287	0.0411	0.0433	0.0498	0.0531	0.0367
Ours R & EW	0.0222	0.0239	**0.0157**	**0.0180**	**0.0356**	**0.0302**	**0.0212**	**0.0348**	**0.0252**

TABLE IX: Results on CTU dataset for landmark localization with different augmentation methods, when trained on the DeepFashion (DF) dataset (top) and in the CTU dataset (bottom). The values represent the normalized error (NE). Best results are marked in bold EW parameters: $\alpha = 200$, $\sigma = 10$.

Methods	Category	
	top-3	top-5
WTBI [50]	43.73	66.26
DARN [4]	59.48	79.58
FashionNet [2]	82.58	90.17
Lu et al. [5]	86.72	92.51
Corbière et al. [6]	86.30	93.80
Wang et al. [25]	90.99	95.78
Liu and Lu [26]	**91.16**	**96.12**
Ours	89.02	94.80
Ours R	89.57	95.09
Ours R & EW	89.63	95.10

TABLE X: Results on DeepFashion dataset for clothing classification and attribute prediction values are in %. Best results are marked in bold

CTU categories	**DeepFashion categories**
bluse	Blouse
hoody	Hoodie, Sweater
pants	Jeans, Jeggins, Joggers, Leggins
polo	Tee, Button-Down
polo-long	Button-Down, Henley, Jacket
skirt	Skirt
tshirt	Tee
tshirt-long	Cardigan, Sweater, Tee

TABLE XI: Mapping of clothing categories between the DeepFashion dataset and the CTU dataset.

229–245.

[4] J. Huang, R. Feris, Q. Chen, and S. Yan, "Cross-Domain Image Retrieval with a Dual Attribute-Aware Ranking Network," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1062–1070.

[5] Y. Lu, A. Kumar, S. Zhai, Y. Cheng, T. Javidi, and R. Feris, "Fully-Adaptive Feature Sharing in Multi-Task Networks with Applications in Person Attribute Classification," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1131–1140.

[6] C. Corbière, H. Ben-Younes, A. Ramé, and C. Ollion, "Leveraging Weakly Annotated Data for Fashion Image Retrieval and Label Prediction," in *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2017, pp. 2268–2274.

[7] Y. Ge, R. Zhang, L. Wu, X. Wang, X. Tang, and P. Luo, "DeepFashion2: A Versatile Benchmark for Detection, Pose Estimation, Segmentation and Re-Identification of Clothing Images," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. [Online]. Available: http://arxiv.org/abs/1901.07973

[8] Y. Ma, J. Jia, S. Zhou, J. Fu, Y. Liu, and Z. Tong, "Towards Better Understanding the Clothing Fashion Styles: A Multimodal Deep Learning Approach," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)*, 2017, pp. 38–44. [Online]. Available: https://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14561

[9] X. Han, Z. Wu, Y.-G. Jiang, and L. S. Davis, "Learning Fashion Compatibility with Bidirectional LSTMs," in *Proceedings of the 25th ACM International Conference on Multimedia*, ser. MM '17. New York, NY, USA: ACM, 2017, pp. 1078–1086. [Online]. Available: http://doi.acm.org/10.1145/3123266.3123394

[10] M. H. Kiapour, X. Han, S. Lazebnik, A. C. Berg, and T. L. Berg, "Where to Buy It: Matching Street Clothing Photos in Online Shops," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 3343–3351.

[11] B. Willimon, I. Walker, and S. Birchfield, "A new approach to clothing classification using mid-level layers," in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 4271–4278.

[12] A. Ramisa, G. Alenyà, F. Moreno-Noguer, and C. Torras, "FINDDD: A fast 3D descriptor to characterize textiles for robot manipulation," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 11 2013, pp. 824–830.

[13] Y. Li, C. Chen, and P. K. Allen, "Recognition of

deformable object category and pose," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 5558–5564.

[14] Y. Li, Y. Wang, M. Case, S. Chang, and P. K. Allen, "Real-time pose estimation of deformable objects using a volumetric approach," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 1046–1052.

[15] J. Stria and V. Hlavác, "Classification of Hanging Garments Using Learned Features Extracted from 3D Point Clouds," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 5307–5312.

[16] A. Ramisa, G. Alenyà, F. Moreno-Noguer, and C. Torras, "A 3D descriptor to detect task-oriented grasping points in clothing," *Pattern Recognition*, vol. 60, pp. 936–948, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0031320316301558

[17] E. Corona, G. Alenyà, A. Gabas, and C. Torras, "Active garment recognition and target grasping point detection using deep learning," *Pattern Recognition*, vol. 74, pp. 629–641, 2018. [Online]. Available: https://doi.org/10.1016/j.patcog.2017.09.042

[18] A. Doumanoglou, J. Stria, G. Peleka, I. Mariolis, V. Petrík, A. Kargakos, L. Wagner, V. Hlaváč, T. Kim, and S. Malassiotis, "Folding Clothes Autonomously: A Complete Pipeline," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1461–1478, 2016.

[19] I. Garcia-Camacho, M. Lippi, M. C. Welle, H. Yin, R. Antonova, A. Varava, J. Borras, C. Torras, A. Marino, G. Alenyà *et al.*, "Benchmarking bimanual cloth manipulation," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1111–1118, 2020.

[20] L. Sun, G. Aragon-Camarasa, S. Rogers, R. Stolkin, and J. P. Siebert, "Single-shot clothing category recognition in free-configurations with application to autonomous clothes sorting," *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 6699–6706, 2017.

[21] A. Doumanoglou, A. Kargakos, T. Kim, and S. Malassiotis, "Autonomous active recognition and unfolding of clothes using random decision forests and probabilistic planning," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 987–993.

[22] A. Doumanoglou, T.-K. Kim, X. Zhao, and S. Malassiotis, "Active Random Forests: An Application to Autonomous Unfolding of Clothes," in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., vol. 8693. Cham: Springer International Publishing, 2014, pp. 644–658.

[23] L. Wagner, D. Krejcová, and V. Smutny, "Ctu color and depth image dataset of spread garments," *Center for Machine Perception, Czech Technical University, Tech. Rep. CTU-CMP-2013-25*, 2013.

[24] S. Yan, Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, "Unconstrained Fashion Landmark Detection via Hierarchical Recurrent Transformer Networks," in *Proceedings of the 25th ACM International Conference on Multimedia*, ser. MM '17. New York, NY, USA: ACM, 2017, pp. 172–180. [Online]. Available: http://doi.acm.org/10.1145/3123266.3123276

[25] W. Wang, Y. Xu, J. Shen, and S.-C. Zhu, "Attentive Fashion Grammar Network for Fashion Landmark Detection and Clothing Category Classification," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4271–4280.

[26] J. Liu and H. Lu, "Deep Fashion Analysis with Feature Map Upsampling and Landmark-Driven Attention," in *Computer Vision - {ECCV} 2018 Workshops - Munich, Germany, September 8-14, 2018, Proceedings, Part {III}*, vol. 11131 LNCS, 2018, pp. 30–36. [Online]. Available: https://doi.org/10.1007/978-3-030-11015-4{_}4

[27] K. Hamajima and M. Kakikura, "Planning strategy for unfolding task of clothes - isolation of clothes from washed mass," in *Proceedings of the 35th SICE Annual Conference. International Session Papers*, 1996, pp. 1237–1242.

[28] P. Jiménez, "Visual grasp point localization, classification and state recognition in robotic manipulation of cloth: An overview," *Robotics and Autonomous Systems*, vol. 92, pp. 107–125, 2017. [Online]. Available: http://dx.doi.org/10.1016/j.robot.2017.03.009

[29] L. Sun, S. Rogers, G. Aragon-Camarasa, and J. P. Siebert, "Recognising the clothing categories from free-configuration using Gaussian-Process-based interactive perception," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 2464–2470.

[30] Y. Kita, F. Saito, and N. Kita, "A deformable model driven visual method for handling clothes," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. 2004*, vol. 4, 4 2004, pp. 3889–3895 Vol.4.

[31] I. Mariolis, G. Peleka, A. Kargakos, and S. Malassiotis, "Pose and category recognition of highly deformable objects using deep learning," in *2015 International Conference on Advanced Robotics (ICAR)*, 2015, pp. 655–662.

[32] C. Kampouris, I. Mariolis, G. Peleka, E. Skartados, A. Kargakos, D. Triantafyllou, and S. Malassiotis, "Multi-sensorial and explorative recognition of garments and their material properties in unconstrained environment," in *IEEE International Conference on Robotics and Automation*, 2016, pp. 1656–1663.

[33] A. Gabas, E. Corona, G. Alenyà, and C. Torras, "Robot-Aided Cloth Classification Using Depth Information and CNNs," in *Articulated Motion and Deformable Objects*, F. J. Perales and J. Kittler, Eds. Cham: Springer International Publishing, 2016, pp. 16–23.

[34] B. Willimon, S. Birchfield, and I. Walker, "Model for unfolding laundry using interactive perception," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2011, pp. 4871–4876.

[35] G. Alenyà Ribas, A. Ramisa Ayats, F. Moreno-Noguer, and C. Torras, "Characterization of textile grasping experiments," in *Proceedings of the 2012 ICRA Workshop on Conditions for Replicable Experiments and Performance Comparison in Robotics Research*, 2012, pp. 1–6.

[36] Y. Kita, T. Ueshiba, E. S. Neo, and N. Kita, "Clothes state recognition using 3D observed data," in *2009 IEEE International Conference on Robotics and Automation*, 2009, pp. 1220–1225.

[37] P. Y. Simard, D. Steinkraus, and J. C. Platt, "Best practices for convolutional neural networks applied to visual document analysis," in *Seventh International Conference on Document Analysis and Recognition, 2003. Proceedings.*, 8 2003, pp. 958–963.

[38] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *Journal of Big Data*, 2019. [Online]. Available: https://doi.org/10.1186/s40537-019-0197-0

[39] A. W. Moore, "Efficient Memory-based Learning for Robot Control," Tech. Rep., 1990.

[40] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in *International Conference on Learning Representations*, 2015.

[41] J. Wang, W. Liu, L. Ma, H. Chen, and L. Chen, "IORN: An Effective Remote Sensing Image Scene Classification Framework," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 11, pp. 1695–1699, 11 2018.

[42] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial Transformer Networks," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 2017–2025. [Online]. Available: http://papers.nips.cc/paper/5854-spatial-transformer-networks.pdf

[43] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation Networks," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7132–7141.

[44] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds. Cham: Springer International, 2015, pp. 234–241.

[45] W. Li, X. Zhu, and S. Gong, "Harmonious Attention Network for Person Re-identification," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2285–2294, 2018.

[46] Y. Zhou, Q. Ye, Q. Qiu, and J. Jiao, "Oriented Response Networks," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 4961–4970.

[47] V. Dumoulin and F. Visin, "A guide to convolution arithmetic for deep learning." *CoRR*, vol. abs/1603.0, 2016. [Online]. Available: http://dblp.uni-trier.de/db/journals/corr/corr1603.html{#}DumoulinV16

[48] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, 2009.

[49] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

[50] H. Chen, A. Gallagher, and B. Girod, "Describing Clothing by Semantic Attributes," in *Computer Vision – ECCV 2012*, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 609–623.