VI. 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 [47]. 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 [48] 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,

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

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 $\mathbf{A}_{\text{spatial}}^L$ and a category attention $\mathbf{A}_{\text{spatial}}^C$.

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

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

The attention branch is designed in a three branch unit; two branches for the spatial attention $\mathbf{A}_{\text{spatial}}^L, \mathbf{A}_{\text{spatial}}^C$ (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.

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

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

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

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 $\mathbf{F} \in$

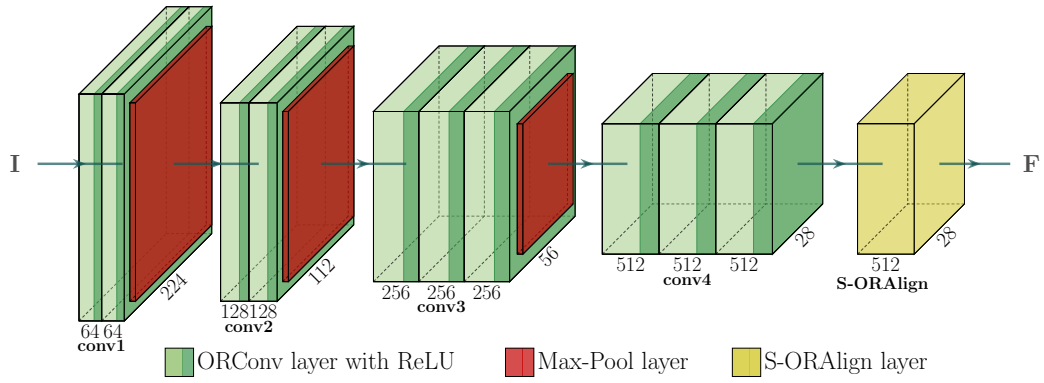


Fig. 4: Structure of the landmark localization branch. Each cuboid represents a feature map of the given layer. The number below a cuboid denotes the number of channels in the feature map and the number on the side denotes the width and height of the feature map. (Best viewed in color)

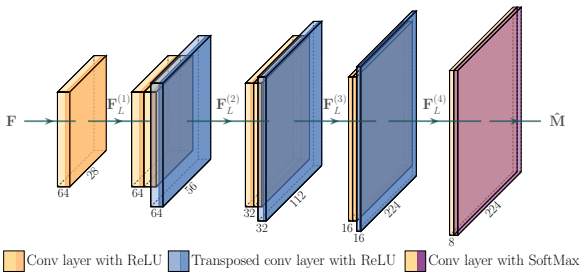


Fig. 5: Structure of the landmark localization branch. Each cuboid represents a feature map of the given layer. The number below a cuboid denotes the number of channels in the feature map and the number on the side denotes the width and height of the feature map. (Best viewed in color)

$\mathbb{R}^{28\times28\times512}$ 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,

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

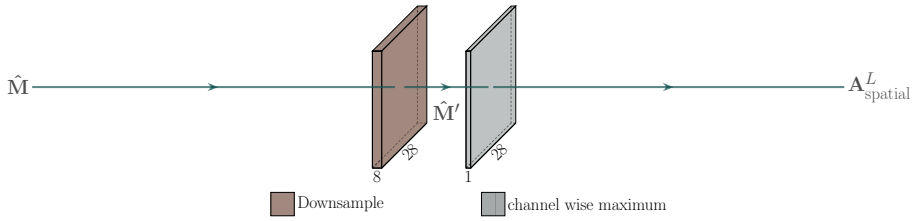
To refine the attention, an additional 1×1 convolution layer is added afterwards. This is motivated by the fact that the spatial and channel attention are not mutually exclusive but with co-occurring complementary relationship [45].

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

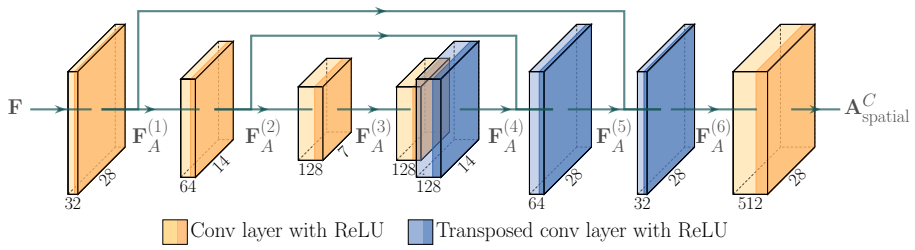
4) Rest of the network: Given $\mathbf{A} \in [-1,1]^{28\times28\times512}$ we weight the S-ORAlign features $\mathbf{F} \in \mathbb{R}^{28\times28\times512}$,

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

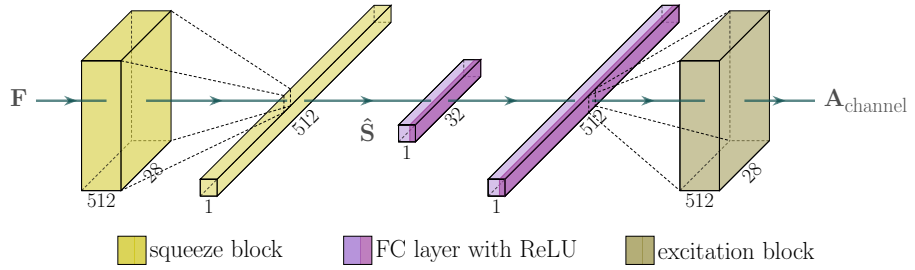
where $\circ$ denotes the Hadamard product and $\mathbf{1}$ is a tensor of ones with size $28\times28\times512$. Hence, features where $\mathbf{A}(\cdot,\cdot,\cdot) \in [-1,0)$ are reduced and features where $\mathbf{A}(\cdot,\cdot,\cdot) \in (0,1]$ are increased. Our attention incorporates semantic information and global information into the network helping to focus on



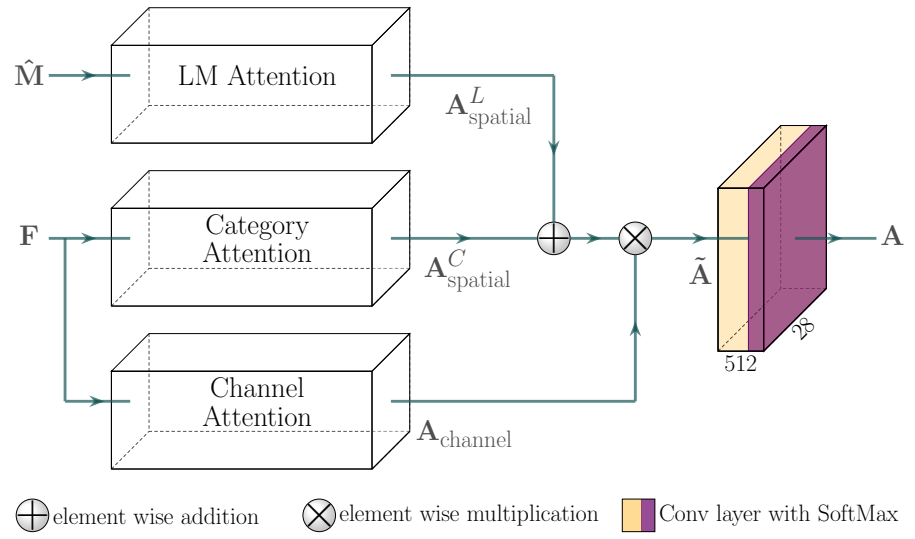
(a) Landmark aware spatial attention



(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)

important regions in the images. The features **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 [49]. 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 [50] 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]–[26] and the clothing category classification to these models [2], [4]–[6], [25], [26], [51]. 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.

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

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 VI-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 VI-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 VI-B1. We perform solely inference with the network. For the evaluation we only consider the categories that are present and mask the rest out.

REFERENCES

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

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: $\alpha = 100$, $\sigma = 10$.

[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. 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-

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 [51]	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.

Identification of Clothing Images," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. arXiv: 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.

[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.

[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*, Nov. 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.

[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.

[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. Smutnỳ, "Ctu color and depth image dataset of spread garments," 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.

[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.

[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.

[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, Apr. 2004, 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.*, Aug. 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.

[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, Nov. 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.

[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] L. Wagner and D. Krejčová, "CTU color and depth image dataset of spread garments.," Center for Machine Perception, K13133 FEE Czech Technical University, Tech. Rep., Sep. 2013.

[47] 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. arXiv: arXiv:1701.01833v2.

[48] V. Dumoulin and F. Visin, "A guide to convolution arithmetic for deep learning.," *CoRR*, vol. abs/1603.0, 2016.

[49] 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.

[50] 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.

[51] 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.