Improving Landmark Localization with Semi-Supervised Learning

Sina Honari[1]*, Pavlo Molchanov[2], Stephen Tyree[2], Pascal Vincent[1,4,5], Christopher Pal[1,3], Jan Kautz[2]
[1]MILA-University of Montreal, [2]NVIDIA, [3]Ecole Polytechnique of Montreal, [4]CIFAR, [5]Facebook AI Research.
[1]{honaris, vincentp}@iro.umontreal.ca,
[2]{pmolchanov, styree, jkautz}@nvidia.com, [3]christopher.pal@polymtl.ca

Abstract

We present two techniques to improve landmark localization in images from partially annotated datasets. Our primary goal is to leverage the common situation where precise landmark locations are only provided for a small data subset, but where class labels for classification or regression tasks related to the landmarks are more abundantly available. First, we propose the framework of sequential multitasking and explore it here through an architecture for landmark localization where training with class labels acts as an auxiliary signal to guide the landmark localization on unlabeled data. A key aspect of our approach is that errors can be backpropagated through a complete landmark localization model. Second, we propose and explore an unsupervised learning technique for landmark localization based on having a model predict equivariant landmarks with respect to transformations applied to the image. We show that these techniques, improve landmark prediction considerably and can learn effective detectors even when only a small fraction of the dataset has landmark labels. We present results on two toy datasets and four real datasets, with hands and faces, and report new state-of-the-art on two datasets in the wild, e.g. with only 5% of labeled images we outperform previous state-of-the-art trained on the AFLW dataset.

1. Introduction

Landmark localization – finding the precise location of specific parts in an image – is a central step in many complex vision problems. Examples include hand tracking [14, 8], gesture recognition [7], facial expression recognition [15], face identity verification [33, 32], and eye gaze tracking [49, 22]. Reliable landmark estimation is often part of the pipeline for sophisticated, robust vision tools. Neural networks have yielded state-of-the art results on numerous landmark estimation problems [36, 13, 41, 45, 39]. However, neural networks generally need to be trained on a large set of labeled data to be robust to the variations in natural images. Landmark labeling is a tedious manual work where precision is important; as a result, few landmark datasets are large enough to train reliable deep neural networks. On the other hand it is much easier to label an image with a single class label rather than the entire set of precise landmarks, and datasets with labels related to—but distinct from—landmark detection are far more abundant.

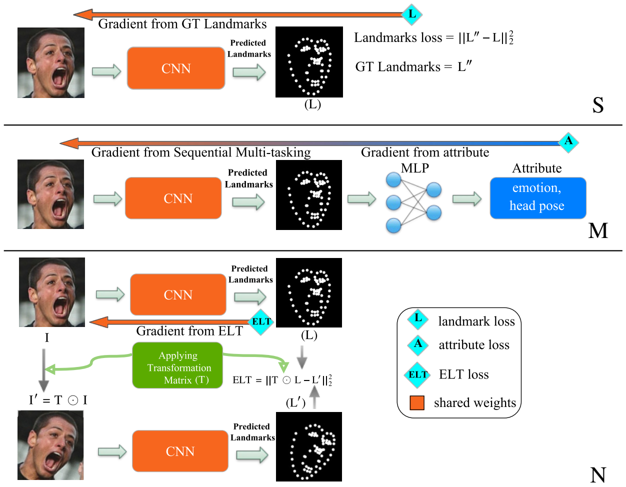

Figure 1: In our approach three sources of gradients are used for learning a landmark localization network, from top to bottom: 1) The gradient from S labeled image-landmark pairs. 2) The gradient from M attribute examples, obtained through sequential multitasking. The first part of the network (a CNN) predicts landmarks with a soft-argmax output layer to make the entire network fully differentiable. The predicted landmarks (as x, y pairs) are then fed into a multi-layer perceptron (MLP) for attribute regression/classification. 3) The gradient received from an unsupervised component of the composite loss which we refer to as an equivariant landmark transformation (ELT) (applied to N images). This loss encourages the model to output landmarks that are equivariant to transformations applied to the image. Importantly, MLP regression and the ELT are applied to the model's predictions and not the ground truth (GT) landmarks, so they can be applied on images that are not labelled with landmarks. Our proposed approach allows efficient training even when $S \ll M \leq N$.

*Part of this work was done when author was at NVIDIA Research

The key elements of our approach are illustrated in Figure 1. The top diagram illustrates a traditional convolutional neural network (CNN) based landmark localization network. The first key element of our work – illustrated in the second diagram of Figure 1, is that we use the indirect supervision of class labels to guide classifiers trained to localize landmarks. The class label can be considered a weak label that sends indirect signals about landmarks. For example, a photo of a hand gesture with the label "waving" likely indicates that the hand is posed with an open palm and spread fingers, signaling a set of reasonable locations for landmarks on the hand. We leverage class labels that are more abundant or more easily obtainable than landmark labels, putting our proposed method in the category of multi-task learning. A common approach [50, 52, 47, 9] to multi-task learning uses a traditional CNN, in which a final common fully-connected (FC) layer feeds into separate branches, each dedicated to the output for a different task. This approach learns shared low-level features across the set of tasks and acts as a regularizer, particularly when the individual tasks have few labeled samples.

There is a fundamental caveat to applying such an approach directly to simultaneous classification and landmark localization tasks, because the two have opposing requirements: classification output needs to be insensitive (invariant) to small deformations such as translations, whereas landmark localization needs to be equivariant to them, i.e., follow them precisely with high sensitivity. To build in invariance, traditional convolutional neural networks for classification problems rely on pooling layers to integrate signals across the input image. However, tasks such as landmark localization or image segmentation require both the global integration of information as well as an ability to retain local, pixel-level details for precise localization. The goal of producing precise landmark localization has thus led to the development of new layers and network architectures such as dilated convolutions [43], stacked what-where auto-encoders [53], recombinator-networks [13], fully-convolutional networks [20], and hyper-columns [11], each preserving pixel-level information. These models have however not been developed with multi-tasking in mind.

Current multi-task architectures [50, 52, 47, 9, 24] predict landmark locations and auxiliary tasks as separate branches, i.e., *in parallel*. In this scenario the auxiliary task is used for partial supervision of landmark localization model. We propose a novel class of neural architectures which force classification predictions to flow through the intermediate step of landmark localization to provide complete supervision during backpropagation.

One of the contributions of our model is to leverage auxiliary classification tasks and data, enhancing landmark localization by backpropagating classification errors through the landmark localization layers of the model. Specifically, we propose a sequential architecture in which the first part of the network predicts landmarks via pixel-level heatmaps, maintaining high-resolution feature maps by omitting pooling layers and strided convolutions. The second part of the network computes class labels using predicted landmark locations. To make the whole network differentiable, we use soft-argmax for extracting landmark locations from pixel-level predictions. Under this model, learning the landmark localizer is more directly influenced by the task of predicting class labels, allowing the classification task to enhance landmark localization learning.

Semi-supervised learning techniques [28, 25, 40, 38] have been used in deep learning to improve classification accuracy with a limited amount of labeled training data. A recently proposed method [18] enforces invariance in class predictions over time and across a variety of data augmentations applied to unlabeled training data. **Our second contribution is to propose and explore an unsupervised learning technique for landmark localization where the model is asked to produce landmark localizations equivariant with respect to a set of transformations applied to the image.** In other words, we transform an image during training and ask the model to produce landmarks that are similarly transformed. Importantly, this technique does not require the true landmark locations, and thus can be applied during semi-supervised training to leverage images with unlabeled landmarks. This element of our work is illustrated in the third diagram of Figure 1. Independently from our work, Thewlis et al. [35] proposed an unsupervised technique for landmark localization, however, the question if it can be used to improve supervised training remains open.

To summarize, in this paper we make the following contributions: 1) We propose a novel multi-tasking neural architecture, which a) predicts landmarks as an intermediate step before classification in order to use the class labels to improve landmark localization, b) uses soft-argmax for a fully-differentiable model in which end-to-end training can be performed, even from examples that do not provide labeled landmarks. 2) We propose an unsupervised learning technique to learn features that are equivariant with respect to transformations applied to the input image. Combining contributions 1) and 2), we propose a robust landmark estimation technique which learns effective landmark predictors while requiring fewer labeled landmarks compared to current approaches. 3) We report state-of-the-art on 300W [27] and AFLW [17] without leveraging any external data.

2. Sequential Multi-Tasking

We refer to the new architecture that we propose for leveraging the attributes to guide the learning of landmark locations as *sequential multi-tasking*. This architecture first predicts the landmark locations and then uses the predicted landmarks as the input to the second part of the network, which performs classification (see Fig. 1-middle). In doing so, we create a bottleneck in the network, forcing it to solve the classification task only through the landmarks. If the goal were to enhance classification, this architecture would have been harmful since such bottlenecks [12] would hurt the flow of information for classification. However, since our goal is landmark localization, this architecture enforces receiving signal from class labels through back-propagation to enhance landmark locations. This architecture benefits from auxiliary tasks that can be efficiently solved relying only on extracted landmark locations without observing the input image.

In order to make the whole pipeline trainable end-to-end, even on examples that do not provide any landmarks, we apply soft-argmax [6] on the output of the last convolutional layer in the landmark prediction model. Specifically, let $M(I)$ be the stack of K two-dimensional output maps produced by the last convolutional layer for a given network input image I. The map associated to the k^{th} landmark will be denoted $M_k(I)$. To obtain a single 2d location $L_k = (x, y)$ for the landmark from $M_k(I)$, we use the following soft-argmax operation:

$$\begin{aligned} L_k(I) &= \text{soft-argmax}(\beta M_k(I)) \\ &= \sum_{i,j} \text{softmax}(\beta M_k(I))_{i,j}(i,j) \end{aligned} \tag{1}$$

where softmax denotes a spatial softmax of the map, i.e. $\text{softmax}(A)_{i,j} = \exp(A_{i,j}) / \sum_{i',j'} \exp(A_{i',j'})$. β controls the temperature of the resulting probability map, and (i, j) iterate over pixel coordinates. In short, soft-argmax computes landmark coordinates $L_k = (x, y)$ as a weighted average of all pixel coordinate pairs (i, j) where the weights are given by a softmax of landmark map M_k.

Predicted landmark coordinates are then fed into the second part of the network for attribute estimation. Having either classification or regression task, the model optimizes

$$Cost\_attr = \begin{cases} -\log P(\mathbb{A} = \tilde{a} | \mathbb{I} = I) & \text{, if classification} \\ |\tilde{a} - a(I)| & \text{, if regression} \end{cases}$$

$P(\mathbb{A} = \tilde{a} | \mathbb{I} = I)$ denotes the probability ascribed by the model to the class $\tilde{a}$ given input image I, as computed by the final classification softmax layer. $\tilde{a}$ denotes the ground truth (GT) and $a(I)$ the predicted attributes in the regression task. Using soft-argmax, as opposed to a simple softmax, the model is fully differentiable through its landmark locations and is trainable end-to-end.

3. Equivariant Landmark Transformation

We propose the following unsupervised learning technique to make the model's prediction consistent with respect to different transformations that are applied to the image. Consider an input image I and the corresponding landmarks $L(I)$ predicted by the network. Now consider a small affine coordinate transformation T. We will use $T \odot \ldots$ to denote the application of such a transformation in coordinate space, whether it is applied to deform a bitmap image or to transform actual coordinates. If we apply this transformation to produce a deformed image $I' = T \odot I$ and compute the resulting landmark coordinates $L(I')$ predicted by the network, they should be very close to the result of applying the transformation on landmark coordinates $L(I)$, i.e., we expect to have $L(T \odot I) \approx T \odot L(I)$. The architecture for this technique, which we call *equivariant landmark transformation (ELT)*, is illustrated in Fig. 1-bottom. Multiple instances of C_T can thus be added to the overall training cost, each corresponding to a different transformation T.

Our entire model is trained end-to-end to minimize the following cost

$$\begin{aligned} Cost = \quad & \tfrac{1}{N} \textstyle\sum_{(I,\tilde{a}) \in \mathcal{D}} \{Cost\_attr \; + \\ & \tfrac{\alpha}{K} \textstyle\sum_{k=1}^{K} \| \underbrace{T \odot L_k(I)}_{\hat{L}_k} - \underbrace{L_k(T \odot I)}_{L'_k} \|_2^2 \} + \\ & \tfrac{\lambda}{SK} \textstyle\sum_{\tilde{L}} \sum_{k=1}^{K} ||\tilde{L}_k - L_k(I)||_2^2 + \gamma ||\mathbb{W}||_2^2, \end{aligned} \tag{2}$$

where $\mathcal{D}$ is the training set containing N pairs $(I, \tilde{a})$ of input image and GT attribute. K is the number of landmarks. $\tilde{L}_k$, $L_k(I)$ and S respectively correspond to the GT, predicted landmarks and the number of images in the train set with labelled landmarks. $\mathbb{W}$ represents the parameters of the model. α, λ, and γ are weights for losses. The first part of the cost is attribute classification or regression and affects the entire network. The second part is the ELT cost and can be applied to any training image, regardless of whether or not it is labeled with landmarks. This cost only affects the first part of the network (Landmark Localization). The third part is the squared Euclidean distance between GT and estimated landmark locations and is used only when landmark labels are provided. This cost only affects the first part of the network. The last cost is ℓ_2-norm on the model's parameters.

4. Experiments

To validate our proposed model, we begin with two toy datasets in Sections 4.1 and 4.2, in order to verify to what extent the class labels can be used to guide the landmark localization regardless of the complexity of the dataset. Later, we evaluate the proposed network on four real datasets: Polish sign-language dataset [16] in Section 4.3, Multi-PIE [10] in Section 4.4, and two datasets in the wild; 300W [27] and AFLW [17] in Sections 4.5 and 4.6. All the models are implemented in Theano [1].

4.1. Shapes dataset

To begin, we use a simple dataset to demonstrate our method's ability to learn consistent landmarks without di-

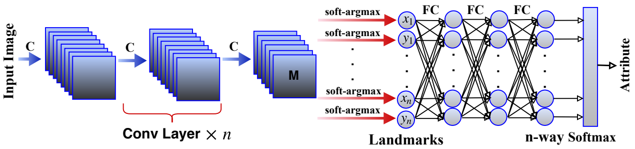

Figure 2: Our basic implementation of the sequential multi-tasking architecture. The landmark localization model is composed a series of conv (C) layers (with no pooling) and a soft-argmax output layer to detect landmarks. $\times n$ indicates repeating conv layer n times without parameter sharing. The detected landmarks are then fed to FC layers for attribute classification.

rect supervision. Images in our Shapes dataset (see Fig. 3 top row for examples) consist of a white triangle and a white square on black background, with randomly sampled size, location, and orientation. The classification task is to identify which shape (triangle or square) is positioned closer to the upper-left corner of the image. We trained a model (as illustrated in Fig. 2) with six convolutional layers using 7×7 kernels, followed by two convolutional layers with 1×1 kernels, then the soft-argmax layer for landmark localization. Predicted landmarks input to two fully connected (FC) layers of size 40 and 2, respectively. The model is trained with *only* the cross-entropy cost on the class label *without* labeled landmarks or the unsupervised ELT cost.

Figure 3 shows the predictions of the trained model on a few samples from the dataset. In the second row, the green shape corresponds to the shape predicted to be the nearest to the upper-left corner, which was learned with 99% accuracy. The red and blue crosses correspond to the first soft-argmax and second soft-argmax landmark localizations, respectively. We observe that the red cross is consistently placed adjacent to the triangle, while the blue cross is near the square. This experiment shows the sequential architecture proposed here properly guides the first part of the network to find meaningful landmarks on this dataset, based solely on the supervision of the related classification task.

4.2. Blocks dataset

Our second toy dataset, Blocks, presents additional difficulty: each image depicts a figure composed of a sequence of five white squares with one white triangle at the head. See Fig. 4-top for all fifteen classes of Block dataset. We split the dataset into train, validation, and test sets, each having 3200 images.

Initially we trained the model with only cross-entropy on the class labels and evaluated the quality of the resulting landmark assignments. Ideally, the model would consistently assign each landmark to a particular block in the sequence from head (the triangle) to tail (the final square). However, in this more complex setting, the model did not predict landmarks consistently across examples. With the addition of the ELT cost, the model learns relatively consistent landmarks between examples *from the same class*, but this consistency does not extend between different classes. Unlike the Shapes dataset—where there was a consistent, if indirect, mapping between landmarks and the classification task—the correspondence among the classification task and landmark identities is more tenuous in the Blocks dataset. Hence, we introduce a labeled set of ground truth (GT) landmark locations and evaluate the landmark localization accuracy by having different percentages of the training set being labelled with landmarks.

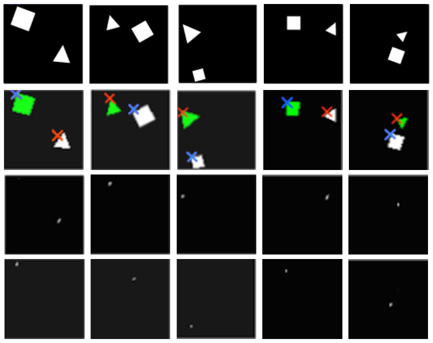

Figure 3: Top row: Sample images from the Shapes dataset. Each 60×60 image contains one square and one triangle with randomly sampled location, size, and orientation. Second row: The two predicted landmarks and the object (in green) closest to the top-left corner classified by network. The *third and fourth* show the first and second landmark feature maps, corresponding closely with the location of triangle and square. (Best viewed in color with zoom.)

Table 1: Error of different architectures on Blocks dataset. The error is reported in pixel space. An error of 1 indicates 1 pixel distance to the target landmark location. The first 4 rows show the results of Seq-MT architecture, as shown in Fig. 2. The 5th and 6th rows show results of Comm-MT, depicted in Fig. 5. The last two rows show the results of Heatmap-MT, depicted in Fig. 6. The results are averaged over five seeds.

	Percentage of Images with Labeled Landmarks					
Model	1%	5%	10%	20%	50%	100%
Seq-MT (L)	8.33	3.95	3.35	1.98	1.19	0.44
Seq-MT (L+A)	8.02	3.45	3.20	1.67	1.05	**0.38**
Seq-MT (L+ELT)	6.42	1.94	1.37	1.16	0.85	
Seq-MT (L+ELT+A)	**6.25**	**1.70**	**1.26**	**1.07**	**0.74**	
Comm-MT (L)	12.89	11.56	10.72	9.39	5.04	3.41
Comm-MT (L+A)	12.28	11.19	10.36	9.01	4.21	2.97
Heatmap-MT (L)	10.09	6.59	5.27	3.82	2.78	2.01
Heatmap-MT (L+A)	9.27	6.35	5.62	3.75	3.14	2.23

Table 1 compares the results using the sequential multi-tasking model in the following scenarios: 1) using only the landmarks (Seq-MT (L)), which is equivalent to training only the first part of the network, 2) using landmarks and attribute labels (Seq-MT (L+A)), which trains the whole network on class labels and the first part of the network on landmarks, 3) using landmarks and the ELT cost (Seq-MT (L+ELT)), which trains only the first part of the network,

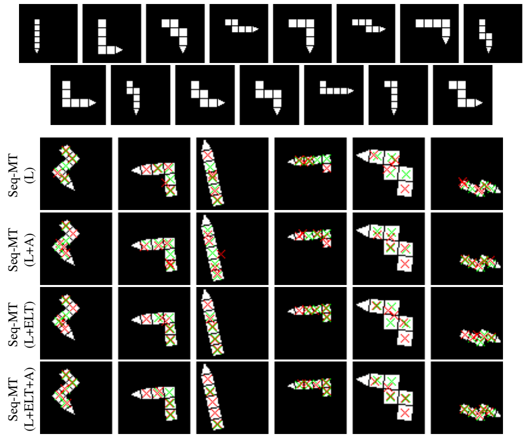

Figure 4: *(top)* The fifteen classes of the Blocks dataset. Each class is composed of five squares and one triangle. To create each 60 × 60 image in the dataset, a random scale, translation, and rotation (up to 360 degrees) is applied to one of the base classes. *(bottom)* Sample landmark prediction on Blocks dataset using sequential multi-tasking models when only 5% of data is labeled with landmarks. Green and red cross show in order GT and predicted landmarks. (Best viewed in color with zoom.)

and 4) using three costs together (Seq-MT (L+ELT+A)).[1] When using the ELT cost (scenarios 3 & 4), we only apply it to images that do not provide GT landmarks to simulate semi-supervised learning [2].

As shown in Table 1, the *Seq-MT (L+A)* improves upon *Seq-MT (L)*, indicating that class labels can be used to guide the landmark locations. By adding the ELT cost, we can improve the results considerably. With *Seq-MT (L+ELT)* better performance is obtained compared to *Seq-MT (L+A)* showing that the unsupervised learning technique can substantially enhance performance. However, the best results are obtained with all costs when using class labels, the ELT and landmark costs. See Fig. 4-bottom for prediction samples when only 5% of the data are labeled with landmarks.

Since our model can be considered as a multi-tasking network, we contrast it with other multi-tasking architectures in the literature. We compare with two architectures: 1) The "common" multi-tasking architecture (Comm-MT) [50, 52, 47, 9] where sub-networks for each task share a common set of initial layers ending in a common fully-connected layer (see Fig. 5).[3] We train two variants of this model, one with only landmarks (*Comm-MT (L)*) and another with landmarks and class labels (*Comm-MT (L+A)*) to see whether the class labels improve landmark localization. 2) Heat-map multi-tasking (Heatmap-MT), where – to avoid pooling layers – we follow the recent trend of maintaining the resolution of feature maps [36, 11, 20, 13] and features detected for landmark localization do not pass through a FC layer. See Fig. 6 for an illustration of this architecture. The heatmaps right before the softmax layer are taken as input to the classification model. Note that this model doesn't have a landmark bottle-neck such as *Seq-MT (L+A)*.

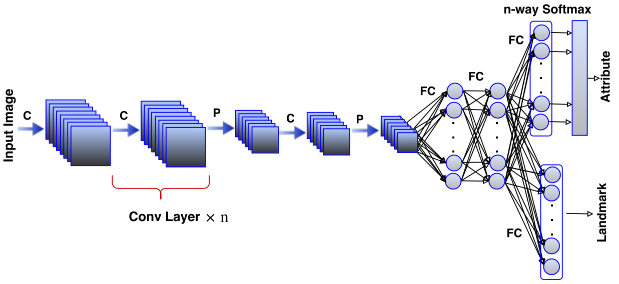

Figure 5: Our implementation of the common multi-tasking (Comm-MT) architecture used in the literature [50, 52, 47, 9]. The model takes an image and applies a series of conv (C) and pooling (P) layers which are then passed to few common (shared) FC layers. The last common FC layer is then connected to two branches (each for a task), one for the classification task and another for the landmark localization.

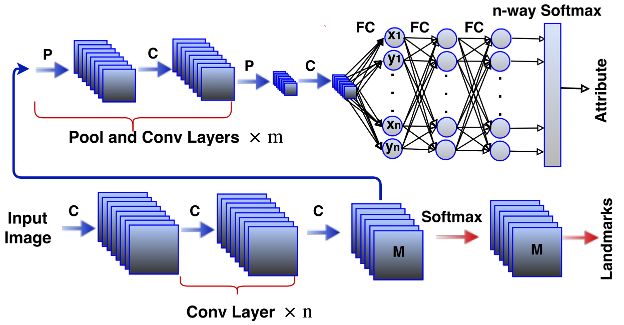

Figure 6: Our implementation of multi-tasking architecture using heatmaps (Heatmap-MT). Landmarks are detected using conv (C) layers without sub-sampling, pooling, or FC layers. A softmax layer is used for landmark prediction in the output layer. Landmark heatmaps right before softmax layer are fed to a series of pool (P) and conv (C) layers which are then passed to FC layers. The last FC layer is fed to softmax for attribute classification.

As shown in Table 1, the Comm-MT approach is performing much worse than our Seq-MT architecture for landmark estimation. A drawback of this architecture is its use of pooling layers which leads to sub-optimal results for landmark estimation. The model trained with extra class information performs better than the model trained only on landmarks. Heatmap-MT also performs worse than Seq-

[1] The set of examples with labeled landmarks is class-balanced.

[2] This is done to avoid unfair advantage of our model compared to other models on examples that provide landmarks. However, the ELT technique can be applied to any image, both with and without labeled landmarks.

[3] We tried other variants such as 1) a model that goes directly from the feature maps that have the same size as input image to the FC layer without any pooling layers and 2) a model that has more pooling layers and goes to a lower resolution before feeding the features to FC layers. Both models achieved worse results. Model 1 suffers from over-parameterization when going to FC layer. Model 2 suffers from loosing track of pooled features' locations since more pooling layers are used.

MT. This is likely due in part to Heatmap-MT using softmax log-likelihood training (which cannot be more accurate than the discretization grid), while Seq-MT uses softargmax training based on real number coordinates. Moreover, in Heatmap-MT the class label is mostly helping when using a low percentage of labeled data, but in Seq-MT it is helping for all percentages of labeled data. We believe this is due to creating a bottle-neck of landmarks before class label prediction, which causes the class labels to impact landmarks more directly through back-propagation.

4.3. Hand pose estimation

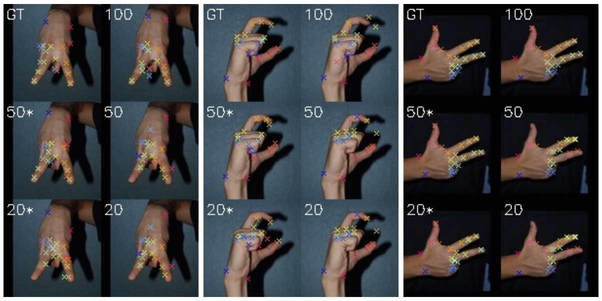

Figure 7: Examples of our model predictions on the test set of the HGR1 dataset [16, 23]. GT represents ground-trust annotations, while numbers 100, 50, and 20 indicate which percentage of the training set with labeled landmarks used for training. Results are computed with Seq-MT (L+ELT+A) model (denoted *) and Seq-MT (L). Best viewed in color with zoom.

Table 2: Performance of architectures on HGR1 hands dataset. The error is Euclidean distance normalized by wrist width. Results are shown as percent; lower is better.

	Percentage of Images with Labeled Landmarks				
Model	5%	10%	20%	50%	100%
Seq-MT (L)	57.6	41.1	32.0	21.4	**15.8**
Seq-MT (L+A)	50.0	38.1	28.1	19.8	16.9
Seq-MT (L+ELT)	43.7	31.5	25.1	**17.7**	
Seq-MT (L+ELT+A)	**38.5**	**30.3**	**24.0**	19.1	
Comm-MT (L)	77.1	62.8	52.7	41.8	35.7
Comm-MT (L+A)	53.4	39.3	35.5	26.9	24.1
Heatmap-MT (L)	66.5	51.9	42.4	30.9	25.5
Heatmap-MT (L+A)	64.8	54.9	43.2	30.5	26.7

Our first experiment on real data is on images of hands captured with color sensors. Most common image datasets with landmarks on hands such as NYU [37] and ICVL [34] do not provide class labels. Also, most of the prior works in landmark estimation for hands are based on depth data [46, 37, 34, 30, 31, 29] whereas estimating from color data is more challenging. We use the Polish hand dataset (HGR1) [16, 23], which provides 898 RGB images with 25 landmarks and 27 gestures from Polish sign language captured with uncontrolled lightning and uncontrolled background from 12 subjects. We divide images by id (with no overlap in subjects between sets) into training set (ids 1 to 8), validation (ids 11 and 12), and test (ids 9 and 10). We end up with 573, 163, and 162 images for training, validation and test sets, respectively. Accuracy of landmark detection on HGR1 dataset is measured by computing average RMSE metric in the image domain for every landmark and normalizing it by wrist width (the Euclidean distance between landmarks #1 and #25). We apply the ELT cost only on the images that do not have GT landmarks. Table 2 shows results for landmark localization on the HGR1 test set. All results are averaged over 5 seeds. We observe: 1) sequential multitasking improves results for most experiments compared to using only landmarks (Seq-MT(L)) or other multi-tasking approaches, 2) the ELT cost significantly improves results for all experiments, and 3) *Seq-MT (L+ELT+A)* compared to *Seq-MT (L)* can achieve the same performance with only half provided landmark labels (see 5%, 10%, 20%). We show examples of landmark prediction with different models in Fig. 7. ELT and attribute classification (A) losses significantly improve results with a smaller fraction of annotated landmarks.

4.4. Multi-PIE dataset

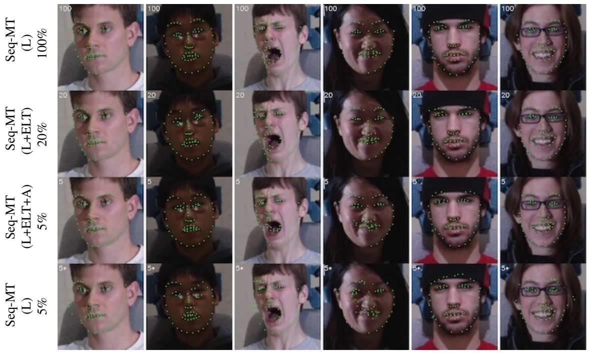

Figure 8: Examples of our model predictions on Multi-PIE [10]. On left you see the percentage of labelled data. We observe close predictions between the top two rows indicating the effectiveness of the proposed ELT cost. Comparison between the last two rows shows the effectiveness of our method with only a small amount of labeled landmarks. Best viewed in color with zoom.

We next evaluate our model on facial landmark datasets. Similar to Hands, most common face datasets including Helen [19], LFPW [3], AFW [56], and 300W [27], only provide landmark locations and no classes. We start with Multi-PIE [10] since it provides, in addition to 68 landmarks, 6 emotions and 15 camera locations. We use these as class labels to guide landmark prediction.[4] We use images from 5 cameras (1 frontal, 2 with ±15 degrees, and 2 with ±30 degrees) and in each case a random illumination

[4] Our research does not involve face recognition, and emotion classes are used only to improve the precision of landmark localization.

Table 3: Performance of different architectures on Multi-PIE dataset. The error is Euclidean distance normalized by eye-centers (as a percent; lower is better). We do not apply ELT cost on the examples that provide GT landmarks.

	Percentage of Images with Labeled Landmarks				
Model	5%	10%	20%	50%	100%
Seq-MT (L)	7.98	7.02	6.28	5.50	**5.09**
Seq-MT (L+A)	7.71	6.91	6.20	5.49	5.12
Seq-MT (L+ELT)	6.69	6.24	5.78	5.27	
Seq-MT (L+ELT+A)	**6.57**	**6.16**	**5.73**	**5.23**	
Comm-MT (L)	9.22	7.93	7.02	6.27	5.71
Comm-MT (L+A)	9.11	8.00	6.92	6.20	5.68
Heatmap-MT (L)	10.83	9.18	8.13	7.00	6.63
Heatmap-MT (L+A)	11.03	9.03	8.15	7.11	6.65

is selected. The images are then divided into subsets by id[5], with ids 1-150 in the training set, ids 151-200 in the valid set, and ids 201-337 in the test set. We end up with 1875, 579, and 1054 images in training, validation, and test sets.

Due to using camera and emotion classes, our classification network has two branches, one for emotion and one for camera, with each branch receiving landmarks as inputs (see Supp. for architecture details). We compare our model with Comm-MT, Heatmap-MT architectures with and without class labels in Table 3. Comparing models, we make the same observation as in Section 4.3 and the best performance is obtained when ELT and classification costs are used jointly, indicating both techniques are affective to get the least error. See some sample predictions in Fig. 8.

4.5. 300W dataset

In order to evaluate our architecture on natural images in the wild we use 300W [27] dataset. This dataset provides 68 landmarks and is composed of 3,148 (337 AFW, 2,000 Helen, and 811 LFPW) and 689 (135 IBUG, 224 LFPW, and 330 Helen) images in the training and test sets, respectively. Similar to RCN [13], we split the training set into 90% (2,834 images) train-set and 10% (314 images) valid-set. Since this dataset does not provide any class label, we can evaluate our model in L and L+ELT cases.

In Table 5-left we compare Seq-MT with other models in the literature. Seq-MT model is outperforming many models including CDM, DRMF, RCPR, CFAN, ESR, SDM, ERT, LBF and CFSS, and is only doing worse than few recent models with complicated architectures, e.g., RCN [13] with multiple branches, RAR [41] with multiple refinement procedure and Lv et. al. [21] with multiple steps. Note that the originality of Seq-MT is not in the specific architecture used for the first part of the network that localizes landmarks, but rather in its multi-tasking architecture (specifically in its usage of the class labels to enhance landmark localization) and also leveraging ELT cost. The landmark localization part of Seq-MT can be replaced with more complex models. To verify this, we use the RCN model [13], with publicly available code, and replace the original softmax layer with a soft-argmax layer in order to apply the ELT cost. We refer to this model as RCN$^+$ and it is trained with these hyperparameters: $\beta = 1.0$, $\alpha = 0.5$, $\gamma = 0$, $\lambda = 1.0$. The result is shown as RCN$^+$(L) when using only landmark cost and RCN$^+$(L+ELT) when using landmark plus ELT cost. On 300W dataset we apply the ELT cost to samples with or without labelled landmarks to observer how much improvement can be obtained when used on all data. We can further reduce RCN error from 5.54 to 5.1 by applying the ELT cost and soft-argmax. This is a new state of the art without any data-augmentation. Also we evaluate accuracy of RCN$^+$(L+ELT) trained without validation set and with early stopping on test set and achieve error of 4.9 - the overall state-of-the-art on this dataset.

In Table 6 we compare Seq-MT with Heatmap-MT and Common-MT on different percentage of labelled landmarks. We also demonstrate the improvement that can be obtained by using RCN$^+$. Note that the ELT cost improves the results when applied to two different landmark localization architectures (Seq-MT, RCN). Moreover, it considerably improves the results on IBUG test-set that contains more difficult examples than the training set. Figure 9 shows the improvement obtained by using ELT cost on some test set samples.

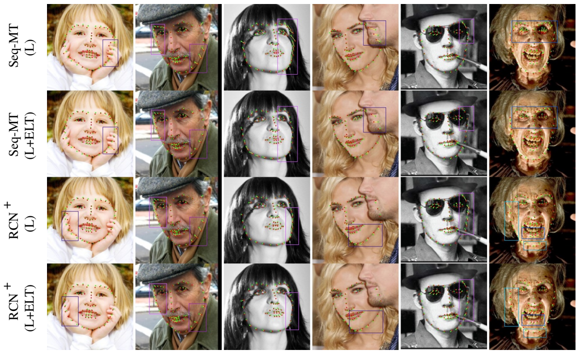

Figure 9: Landmark localization samples on 300W [27] test-set. The green and red dots show GT and model predictions, respectively. The yellow lines show the error. These examples illustrate the improved accuracy obtained by using the ELT cost. The rectangles show the regions that landmarks are mostly improved.

4.6. AFLW dataset

AFLW [17] contains images of 24,386 faces with 19 fiducial landmarks and 3D real-valued head pose information. We use pose as auxiliary task. We split dataset into training, testing sets, with 20,000 and 4,386 images, respectively. Furthermore, we allocate 2,000 images from training set for validation set. We use the same splits as in previous work [26], [21] for direct comparison. We normalize RMSE by face size as in [21]. We evaluate our method

[5] These ids are not personally identifiable information.

Table 4: Comparison of recent models on their training conditions. RAR and Lv et. al [21] initialize their models by pre-trained parameters. TCDCN uses 20,000 extra labelled data. Finally, RAR adds manual samples by occluding images with sunglasses, medical masks, phones, etc to make them robust to occlusion. Similar to RCN, Seq-MT and RCN$^+$ both have an explicit validation set for HP selection and therefore use a smaller training set. Neither use any extra data, either through pre-trained models or explicit external data.

	Model						
Feature	RAR [41]	Lv et. al [21]	TCDCN [51]	CFSS [54]	RCN [13]	Seq-MT / RCN$^+$	RCN$^+$ (L+ELT) (all-train)
Hyper-parameter selection dataset	Test-set	Test-set	Test-set	Test-set	Valid-set	Valid-set	Test-set
Training on entire training set?	Yes	Yes	Yes	Yes	No	No	Yes
Uses extra dataset?	Yes	Yes	Yes	No	No	No	No
Manually augmenting the training set?	Yes	No	No	No	No	No	No
FPS (GPU)	4	83	667	-	545	487 / 545	545

Table 5: Comparison with other SOTA models (as a percent; lower is better). *(left)* Performance of different architectures on 300W test-set using 100% labeled landmarks. The error is Euclidean distance normalized by ocular distance. *(right-top)* Comparison with four other multi-tasking approaches and RCN. For these comparisons, we have implemented the specific architectures proposed in those papers. Error is as in Sections 4.3 and 4.4. *(right-bottom)* Comparison of different architectures on AFLW test set. The error is Euclidean distance normalized by face size.

300W Dataset

Model	**Common**	**IBUG**	**Fullset**
CDM [44]	10.10	19.54	11.94
DRMF [2]	6.65	19.79	9.22
RCPR [4]	6.18	17.26	8.35
CFAN [48]	5.50	16.78	7.69
ESR [5]	5.28	17.00	7.58
SDM [42]	5.57	15.40	7.50
ERT [5]			6.40
LBF [26]	4.95	11.98	6.32
CFSS [54]	4.73	9.98	5.76
TCDCN* [51]	4.80	8.60	5.54
RCN [13]	4.70	9.00	5.54
RCN +\ denoising [13]	4.67	8.44	5.41
RAR [41]	**4.12**	8.35	4.94
Lv et. al [21]	4.36	**7.56**	4.99
Heatmap-MT (L)	6.18	13.56	7.62
Comm-MT (L)	5.68	11.04	6.73
Seq-MT (L)	4.93	10.24	5.95
Seq-MT (L+ELT)	4.84	9.53	5.74
RCN$^+$ (L)	4.47	8.47	5.26
RCN$^+$ (L+ELT)	4.34	8.20	5.10
RCN$^+$ (L+ELT) (all-train)	4.20	7.78	**4.90**

Model	**Multi-PIE**		**HGR1**
Percent Labelled	5%	100%	100%
MT-DCNN [47](L+A)	11.13	7.60	20.87
TCDCN [50](L+A)	18.46	10.59	25.85
TCDCN-2 [52](L+A)	10.75	5.83	18.81
MT-Conv [9](L+A)	9.99	8.08	19.20
RCN [13] (L)	7.53	5.78	13.65
RCN+ (L)	6.89	5.04	11.02
RCN+ (L+A)	**6.82**	**4.97**	**10.88**

AFLW Dataset

Model	**Labeled Images**		
	1%	5%	100%
CDM [44]	-	-	5.43
ERT [5]	-	-	4.35
LBF [26]	-	-	4.25
SDM [42]	-	-	4.05
CFSS [54]	-	-	3.92
RCPR [4]	-	-	3.73
CCL [55]	-	-	2.72
Lv et. al [21]	-	-	2.17
RCN$^+$ (L)	2.88	2.17	1.61
RCN$^+$ (L+A)	2.52	2.08	1.60
RCN$^+$ (L+ELT+A)	2.46	**2.03**	**1.59**

Table 6: Performance of different architectures on 300W test-set. The error is Euclidean distance normalized by ocular distance (eye-centers). Error is shown as a percent; lower is better.

		Percentage of Images with Labeled Landmarks				
	Model	5%	10%	20%	50%	100%
Fullset	Heatmap-MT (L)	13.47	11.68	9.85	8.18	7.62
	Comm-MT (L)	16.73	9.66	8.61	7.39	6.73
	Seq-MT (L)	9.82	8.30	7.26	6.28	5.95
	Seq-MT (L+ELT)	8.23	7.28	6.62	6.10	5.74
	RCN$^+$ (L)	7.26	6.48	5.91	5.52	5.26
	RCN$^+$ (L+ELT)	**7.22**	**6.32**	**5.88**	**5.45**	**5.10**
IBUG	Heatmap-MT (L)	26.36	22.77	18.46	14.94	13.56
	Comm-MT (L)	28.64	16.17	14.56	12.16	11.04
	Seq-MT (L)	18.74	16.21	13.41	11.20	10.24
	Seq-MT (L+ELT)	14.68	12.73	11.39	10.37	9.53
	RCN$^+$ (L)	15.36	12.74	11.82	10.12	8.47
	RCN$^+$ (L+ELT)	**12.54**	**10.35**	**9.56**	**8.67**	**8.20**

on RCN$^+$ trained with ELT cost and head pose regression cost and obtain a new state of the art of 1.59 with 27% relative improvement. See comparison with other models in Table 5-righ-bottom. We also evaluate our method with only 180 (1%) or 900 (5%) images of labeled landmarks. Under these settings we get significant improvement with semi-supervised learning. With only 5% of labeled data our method outperforms the previous state of the art methods.

4.6.1 Comparison with other techniques

In Table 5 we compare with recent models proposed for landmark localization and in Table 4 we evaluate their training conditions. RAR, TCDCN, Lv et. al., and CFSS do not use an explicit validation set. This makes comparison with these models more difficult for two reasons: 1) These models do hyper-parameter (HP) selection on the test set, which makes them overfit on the test set; and 2) Their effective training size is bigger. When we use the entire training set (row *RCN$^+$ (L+ELT) (all-train)* in Table 5-left we report new SOTA on 300W dataset. The first three models use extra datasets, either through pre-trained models (RAR, Lv et. al.) or additional labeled data (TCDCN), while we do not leverage any extra data. Finally, our method is 136 and 6.5 times faster than RAR and Lv et. al methods.

5. Conclusion

We presented a new architecture and training procedure for semi-supervised landmark localization. Our contributions are twofold; We first proposed an unsupervised technique that leverages equivariant landmark transformation without requiring labeled landmarks. In addition we developed an architecture to improve landmark estimation using auxiliary attributes such as class labels by backpropagating errors through the landmark localization components of the model. Experiments show that these achieve high accuracy with far fewer labeled landmark training data in tasks of landmark location for hands and faces. We achieve new state of the art performance on public benchmark datasets for fiducial points in the wild, 300W and AFLW.

Acknowledgment

We would like to thank Compute Canada and Calcul Quebec for providing computational resources. This work was partially funded by NVIDIA's NVAIL program.

References

[1] R. Al-Rfou, G. Alain, A. Almahairi, C. Angermueller, et al. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, abs/1605.02688, May 2016.

[2] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic. Robust discriminative response map fitting with constrained local models. In *CVPR*, pages 3444–3451, 2013.

[3] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar. Localizing parts of faces using a consensus of exemplars. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12):2930–2940, 2013.

[4] X. Burgos-Artizzu, P. Perona, and P. Dollár. Robust face landmark estimation under occlusion. In *ICCV*, pages 1513–1520, 2013.

[5] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. In *IJCV*, 107(2):177–190, 2014.

[6] O. Chapelle and M. Wu. Gradient descent optimization of smoothed information retrieval metrics. *Information retrieval*, 13(3):216–235, 2010.

[7] N. Dardas, Q. Chen, N. D. Georganas, and E. M. Petriu. Hand gesture recognition using bag-of-features and multi-class support vector machine. In *Haptic Audio-Visual Environments and Games (HAVE), 2010 IEEE International Symposium on*, pages 1–5. IEEE, 2010.

[8] D. Datcu and S. Lukosch. Free-hands interaction in augmented reality. In *Proceedings of the 1st symposium on Spatial user interaction*, pages 33–40. ACM, 2013.

[9] T. Devries, K. Biswaranjan, and G. W. Taylor. Multi-task learning of facial landmarks and expression. In *Computer and Robot Vision (CRV), 2014 Canadian Conference on*, pages 98–103. IEEE, 2014.

[10] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-PIE. *Image and Vision Computing*, 28(5):807–813, 2010.

[11] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 447–456, 2015.

[12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[13] S. Honari, J. Yosinski, P. Vincent, and C. Pal. Recombinator networks: Learning coarse-to-fine feature aggregation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5743–5752, 2016.

[14] K. Hu, S. Canavan, and L. Yin. Hand pointing estimation for human computer interaction based on two orthogonal-views. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 3760–3763. IEEE, 2010.

[15] S. E. Kahou, C. Pal, X. Bouthillier, P. Froumenty, Ç. Gülçehre, R. Memisevic, P. Vincent, A. Courville, Y. Bengio, R. C. Ferrari, et al. Combining modality specific deep neural networks for emotion recognition in video. In *Proceedings of the 15th ACM on International conference on multimodal interaction*, pages 543–550. ACM, 2013.

[16] M. Kawulok, J. Kawulok, J. Nalepa, and B. Smolka. Self-adaptive algorithm for segmenting skin regions. *EURASIP Journal on Advances in Signal Processing*, 2014(1):170, 2014.

[17] M. Köstinger, P. Wohlhart, P. M. Roth, and H. Bischof. Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *In: Benchmarking Facial Image Analysis Technologies (ICCV Workshop*, 2011.

[18] S. Laine and T. Aila. Temporal ensembling for semi-supervised learning. In *International Conference on Learning Representation*, 2017.

[19] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang. Interactive facial feature localization. In *European Conference on Computer Vision*, pages 679–692. Springer, 2012.

[20] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[21] J. Lv, X. Shao, J. Xing, C. Cheng, and X. Zhou. A deep regression architecture with two-stage re-initialization for high performance facial landmark detection. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[22] K. A. F. Mora and J.-M. Odobez. Gaze estimation from multimodal kinect data. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 25–30. IEEE, 2012.

[23] J. Nalepa and M. Kawulok. Fast and accurate hand shape classification. In *International Conference: Beyond Databases, Architectures and Structures*, pages 364–373. Springer, 2014.

[24] R. Ranjan, V. M. Patel, and R. Chellappa. Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. *arXiv preprint arXiv:1603.01249*, 2016.

[25] A. Rasmus, M. Berglund, M. Honkala, H. Valpola, and T. Raiko. Semi-supervised learning with ladder networks. In *Advances in Neural Information Processing Systems*, pages 3546–3554, 2015.

[26] S. Ren, X. Cao, Y. Wei, and J. Sun. Face alignment at 3000 fps via regressing local binary features. In *CVPR*, pages 1685–1692, 2014.

[27] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. 300 faces in-the-wild challenge: The first facial landmark localization challenge. In *ICCV Workshop*, pages 397–403, 2013.

[28] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training GANs. In *Advances in Neural Information Processing Systems*, pages 2226–2234, 2016.

[29] A. Sinha, C. Choi, and K. Ramani. Deephand: Robust hand pose estimation by completing a matrix imputed with deep features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4150–4158, 2016.

[30] S. Sridhar, A. Oulasvirta, and C. Theobalt. Interactive markerless articulated hand motion tracking using RGB and depth

data. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2456–2463, 2013.

[31] X. Sun, Y. Wei, S. Liang, X. Tang, and J. Sun. Cascaded hand pose regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 824–832, 2015.

[32] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. In *Advances in neural information processing systems*, pages 1988–1996, 2014.

[33] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1701–1708, 2014.

[34] D. Tang, H. Jin Chang, A. Tejani, and T.-K. Kim. Latent regression forest: Structured estimation of 3d articulated hand posture. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3786–3793, 2014.

[35] J. Thewlis, H. Bilen, and A. Vedaldi. Unsupervised learning of object landmarks by factorized spatial embeddings. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[36] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler. Efficient object localization using convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 648–656, 2015.

[37] J. Tompson, M. Stein, Y. Lecun, and K. Perlin. Real-time continuous pose recovery of human hands using convolutional networks. *ACM Transactions on Graphics (ToG)*, 33(5):169, 2014.

[38] C. Wan, T. Probst, L. Van Gool, and A. Yao. Crossing nets: Combining gans and vaes with a shared latent space for hand pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 680–689, 2017.

[39] W. Wang, S. Tulyakov, and N. Sebe. Recurrent convolutional face alignment. In *Asian Conference on Computer Vision*, pages 104–120, 2016.

[40] J. Weston, F. Ratle, H. Mobahi, and R. Collobert. Deep learning via semi-supervised embedding. In *Neural Networks: Tricks of the Trade*, pages 639–655. Springer, 2012.

[41] S. Xiao, J. Feng, J. Xing, H. Lai, S. Yan, and A. Kassim. Robust facial landmark detection via recurrent attentive-refinement networks. In *European Conference on Computer Vision*, pages 57–72. Springer, 2016.

[42] X. Xiong and F. De la Torre. Supervised descent method and its applications to face alignment. In *CVPR*, pages 532–539, 2013.

[43] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. In *International Conference on Learning Representation*, 2016.

[44] X. Yu, J. Huang, S. Zhang, W. Yan, and D. Metaxas. Pose-free facial landmark fitting via optimized part mixtures and cascaded deformable shape model. In *ICCV*, pages 1944–1951, 2013.

[45] X. Yu, F. Zhou, and M. Chandraker. Deep deformation network for object landmark localization. In *European Conference on Computer Vision*, pages 52–70. Springer, 2016.

[46] S. Yuan, G. Garcia-Hernando, B. Stenger, T.-K. Kim, et al. Depth-based 3d hand pose estimation: From current achievements to future goals. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[47] C. Zhang and Z. Zhang. Improving multiview face detection with multi-task deep convolutional neural networks. In *Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on*, pages 1036–1041. IEEE, 2014.

[48] J. Zhang, S. Shan, M. Kan, and X. Chen. Coarse-to-fine auto-encoder networks (cfan) for real-time face alignment. In *ECCV*, pages 1–16, 2014.

[49] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling. Appearance-based gaze estimation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4511–4520, 2015.

[50] Z. Zhang, P. Luo, C. Loy, and X. Tang. Facial landmark detection by deep multi-task learning. In *ECCV*, pages 94–108, 2014.

[51] Z. Zhang, P. Luo, C. C. Loy, and X. Tang. Learning deep representation for face alignment with auxiliary attributes. In *PAMI*, 2015.

[52] Z. Zhang, P. Luo, C. C. Loy, and X. Tang. Learning deep representation for face alignment with auxiliary attributes. *IEEE transactions on pattern analysis and machine intelligence*, 38(5):918–930, 2016.

[53] J. J. Zhao, M. Mathieu, R. Goroshin, and Y. LeCun. Stacked what-where auto-encoders. In *International Conference on Learning Representation - Workshop Track*, 2016.

[54] S. Zhu, C. Li, C. C. Loy, and X. Tang. Face alignment by coarse-to-fine shape searching. In *CVPR*, pages 4998–5006, 2015.

[55] S. Zhu, C. Li, C.-C. Loy, and X. Tang. Unconstrained face alignment via cascaded compositional learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[56] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *CVPR*, pages 2879–2886, 2012.

Supplementary Information for
Improving Landmark Localization with Semi-Supervised Learning

S.5.1. Comparison on MTFL dataset

In table S1 we compare with other models on MTFL [50] dataset which provides 5 landmarks on facial images: eye-centers, nose tip, mouth corners. We follow the same protocol as [13] for comparison, where we use train and valid sets of 9,000 and 1,000 images, respectively. We test our model on AFLW and AFW subsets, with 29,995 and 337 images, that were re-annotated with 5 landmarks. For the $L + A$ case we use the head-pose which is categorized into one of the five cases: right profile, right, frontal, left, left profile. Other attribute labels, e.g. gender and wearing glasses, cannot be determined from such few landmarks and therefore are not useful in our proposed semi-supervised learning of landmarks.

Table S1: Results on MTFL test sets for 100% labelled data

	Model					Our	
	ESR	RCPR	SDM	TCDCN	RCN	RCN+(L)	RCN+(L+A)
AFLW	12.4	11.6	8.5	8.0	5.6	5.22	**5.02**
AFW	10.4	9.3	8.8	8.2	5.36	5.13	**5.08**

S.5.2. Selecting auxiliary labels for semi-supervised learning

The impact of an attribute on the landmark in sequential training depends on the amount of informational overlap between the attribute and the landmarks. We suggest to measure the normalized mutual information adjusted to randomness (Adjusted Mutual Information (AMI)), as a selection heuristic, prior to applying our method. AMI ranges from 0 to 1 and indicates the fraction of statistical overlap. We compute for each attribute its AMI with all landmark coordinates.

On Multi-PIE we got AMI(x;y) = 0.045, indicating a low mutual information between coordinates x and y. We therefore compute AMI for attribute (A) and every landmark (as x,y pair) by discretizing every variable uniformly under assumption of coordinate independence: AMI(A;x,y) = AMI(A;x) + AMI(A;y). Every variable is uniformly discretized to have 20 levels at most. Finally we measure averaged mutual information between an attribute and the set of landmarks as

$$\frac{1}{N \times L} \sum_{n \in N} \sum_{x_n \in X_n, y_n \in Y_n} AMI(A_n; x_n) + AMI(A_n; y_n)$$

where N and L indicate the number of samples and landmarks, X_n and Y_n indicate the set of x and y landmark coordinates per sample n. In Table S2 we observe that hand gesture labels and head pose regression are among the most effective attributes for our method. There is little mutual information between wearing glasses and landmarks, indicating lack of usefulness of this attribute for our semi-supervised setting.

Table S2: Mutual Information between all landmarks and each attribute

Dataset		MultiPIE			HGR1
Attribute	Random	Emotion	Camera	Identity	Gesture Label
AMI, mean	.000	.098	.229	.049	.559
AMI, max	.006	.229	.493	.088	.669

Dataset		AFLW	MTFL	
Attribute	Random	Pose Regression	Glasses	Pose Classification
AMI, mean	.000	.536	.002	.069
AMI, max	.006	.576	.003	.222

The attributes that are mostly useful yield a high accuracy, or low error, if we just train a neural network that takes only ground truth landmarks as input and predicts the attribute. This indicates that by relying only on landmarks we can get high accuracy for those attributes. In Table S3 we compare the attribute prediction accuracy from the proposed Seq-MT model with a case when we do such prediction from GT landmarks. Prediction from GT landmarks always outperforms the one of Seq-MT. This indicates that in our semi-supervised setting, where we have few labelled landmarks, by improving the predicted locations of landmarks, both attribute and landmarks error would reduce.

Table S3: Attribute classification accuracy (MultiPIE, HGR1)—higher is better—or prediction error (AFLW)—lower is better—from GT & estimated landmarks.

	MultiPIE		HGR1	AFLW
Attribute	Camera	Emotion	Label	Pose Error
From GT Landmarks	99.54 ↑	88.21 ↑	91.7 ↑	4.98 ↓
Best Seq-MT Attr. Predict.	98.96	86.48	79.1	5.10

S.5.3. Comparison of softmax and soft-argmax

Heatmap-MT(L) and Seq-MT(L) have the same architectures but use different loss functions (softmax vs. soft-argmax). RCN(L) and RCN+(L) also only differ in their loss function. When comparing these models in Tables 1, 2, 3, 5, and 6 soft-argmax outperforms soft-max. To further examine these two losses we replace soft-max with soft-argmax in Heatmap-MT and show the results in Table S4. Comparing the results in Table S4 with Tables 2 and 3, we observe improved performance of landmark localization using soft-armgax. In soft-max the model cannot be more accurate than the number of elements in the grid, since softmax does a classification over the pixels. However, in soft-argmax the model can regress to any real number and hence can get more accurate results. We believe this is the reason behind its better performance.

Table S4: Results on Heatmap-MT (L+A) comparing softmax with soft-argmax.

Dataset		5%	10%	20%	50%	100%
Multi-PIE	softmax	11.03	9.03	8.15	7.11	6.65
	soft-argmax	**8.00**	**7.06**	**6.29**	**5.49**	**5.14**
HGR1	softmax	64.8	54.9	43.2	30.5	26.7
	soft-argmax	**56.88**	**42.79**	**33.07**	**22.5**	**18.8**

S.5.4. Supplementary results on Multi-PIE dataset

Although the focus of this paper is on improving landmark localization, in order to observe the impact of each multi-tasking approach on the attribute classification accuracy, we report the classification results on emotion in Table S5 and on camera in Table S6. Results show that the classification accuracy improves by providing more labeled landmarks, despite having the number of *(image, class label)* pairs unchanged. It indicates that improving landmark localization can directly impact the classification accuracy. Landmarks are especially more helpful in emotion classification. On camera classification, the improvement is small and all models are getting high accuracy. Another observation is that Heatmap-MT performs better on classification tasks compared to the other two multi-tasking approaches. We believe this is due to passing more high-level features from the image to the attribute classification network compared to Seq-MT. However, this model is performing worse than Seq-MT on landmark localization. The Seq-MT model benefits from the landmark bottleneck to improve its landmark localization accuracy. In Tables S5 and S6 by adding the ELT cost the classification accuracy improves (in addition to landmarks) indicating the improved performance in landmark localization can enhance classification performance.

Figure S1 provides further localization examples on Multi-PIE dataset.

S.5.5. Supplementary results on hands dataset

In Table S7 we show classification accuracy obtained using different multi-tasking techniques. Similar to the Multi-PIE dataset, we observe increased accuracy by providing more labeled landmarks, showing the classification would benefit directly from landmarks. Also similar to Multi-PIE, we observe better classification accuracy with Heatmap-MT. Comparing Seq-MT models, we observe improved classification accuracy by using the ELT cost. It demonstrates the impact of this component on both landmark localization and classification accuracy.

Table S5: Emotion classification error on Multi-PIE test set. In percent; higher is better.

	Percentage of Images with Labeled Landmarks				
Model	5%	10%	20%	50%	100%
Comm-MT (L+A)	74.67	79.90	83.76	86.37	86.83
Heatmap-MT (L+A)	**85.14**	**87.50**	**86.93**	**88.16**	**87.29**
Seq-MT (L+A)	78.78	82.62	84.69	84.03	84.86
Seq-MT (L+A+ELT)	82.90	84.57	84.85	86.48	

Table S6: Camera classification error on Multi-PIE test set. In percent; higher is better.

	Percentage of Images with Labeled Landmarks				
Model	5%	10%	20%	50%	100%
Comm-MT (L+A)	96.98	97.53	98.30	98.63	98.80
Heatmap-MT (L+A)	**98.46**	**98.99**	**98.99**	**98.98**	**98.98**
Seq-MT (L+A)	97.97	98.31	98.50	98.96	98.92
eq-MT (L+A+ELT)	98.41	98.53	98.47	98.43	

Table S7: Classification error on hands test set. In percent; higher is better.

	Percentage of Images with Labeled Landmarks				
Model	5%	10%	20%	50%	100%
Comm-MT (L+A)	60.86	69.64	69.20	76.03	73.42
Heatmap-MT (L+A)	**83.74**	**87.86**	**87.55**	**90.29**	**89.27**
Seq-MT (L+A)	69.08	70.14	72.26	77.07	75.92
Seq-MT (L+A+ELT)	74.64	75.01	73.90	79.10	

Figure S2 provides further landmark localization examples on hands dataset.

S.5.6. Supplementary results on 300W dataset

In Figure S3 we show the architecture of *RCN* $^+$ used for 300W and AFLW datasets. In Figure S4 we illustrate further samples from 300W dataset. The samples show the improved accuracy obtained in both *Seq-MT* and *RCN* $^+$ by using the ELT loss.

S.5.7. Supplementary results on AFLW dataset

In Table S8 we show pose estimation error using different percentage of labelled data for RCN$^+$ (L+ELT+A) model and compare the results to a model trained to estimate pose from GT landmarks. All models get close results compared to GT model indicating RCN$^+$ (L+ELT+A) can do a reliable pose estimation using a small set of labelled landmarks.

Figure S5 shows some samples on AFLW test set.

S.5.8. Architecture details

The architecture details of Seq-MT model on different datasets can be seen in Tables S11, S12 and S13. Architecture details of Comm-MT and Heatmap-MT for Blocks dataset are shown in Tables S9 and S10. For other dataset,

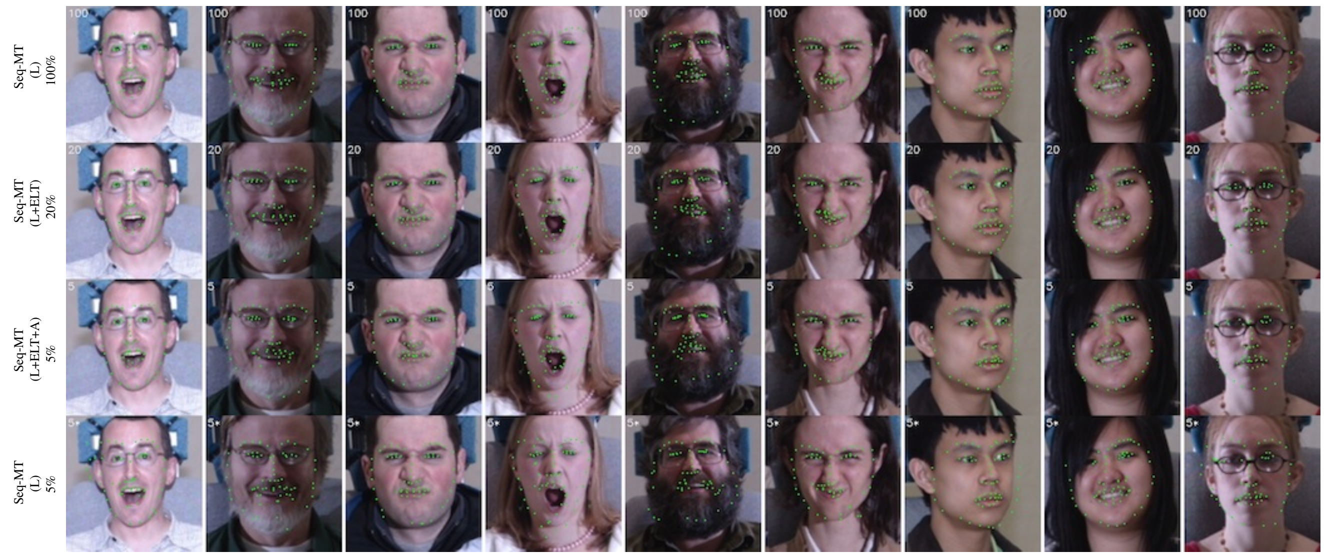

Figure S1: Extra examples of our model predictions on Multi-PIE [10] test set. We observe close predictions by 1) and 2) indicating the effectiveness of our proposed ELT cost even with only a small amount of labeled landmarks. Comparison between 3) and 4) shows the improvement obtained with both the ELT loss and the sequential multitasking architecture when using a small percentage of labeled landmarks. Note that the model trained with ELT loss preserves better the joint distribution over the landmarks even with a small number of labeled landmarks. The last two examples show examples with high errors. Best viewed in color with zoom.

Table S8: Pose degree estimation error on AFLW test set, as average of yaw, pitch, roll values. lower is better.

	Percentage of Images with Labeled Landmarks		
Model	1%	5%	100%
RCN^{+}(L+ELT+A)	5.05	5.01	5.12
GT			4.98

Table S9: Architecture details for Comm-MT Model on Blocks dataset.

<table>
<tr><th colspan="2">Input = $60 \times 60 \times 1$</th></tr>
<tr><td colspan="2">Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME
Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME
Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME
Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME
Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME
Pool 2×2, stride 2
Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME
Pool 2×2, stride 2
Conv $1 \times 1 \times 8$, ReLU, stride 1, SAME
Conv $1 \times 1 \times 8$, ReLU, stride 1, SAME
FC $\#units = 256$, ReLU, dropout-prob=.25
FC $\#units = 256$, ReLU, dropout-prob=.25</td></tr>
<tr><td>Classification branch</td><td>Landmark localization branch</td></tr>
<tr><td>FC $\#units = 15$, Linear
softmax(dim=15)</td><td>FC $\#units = 10$, Linear</td></tr>
</table>

Table S10: Architecture details for Heatmap-MT Model on Blocks datasets.

<table>
<tr><th colspan="2">Input = $60 \times 60 \times 1$</th></tr>
<tr><td colspan="2">Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME
Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME
Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME
Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME
Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME
Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME
Conv $1 \times 1 \times 8$, ReLU, stride 1, SAME
Conv $1 \times 1 \times 5$, ReLU, stride 1, SAME</td></tr>
<tr><td>classification branch</td><td>landmark localization branch</td></tr>
<tr><td>Pool 2×2, stride 2</td><td>—</td></tr>
<tr><td>Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME</td><td>—</td></tr>
<tr><td>Pool 2×2, stride 2</td><td>—</td></tr>
<tr><td>Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME</td><td>—</td></tr>
<tr><td>Pool 2×2, stride 2</td><td>—</td></tr>
<tr><td>Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME</td><td>—</td></tr>
<tr><td>Pool 2×2, stride 2</td><td>—</td></tr>
<tr><td>Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME</td><td>—</td></tr>
<tr><td>FC $\#units = 256$, ReLU, dropout-prob=.25</td><td>—</td></tr>
<tr><td>FC $\#units = 256$, ReLU, dropout-prob=.25</td><td>—</td></tr>
<tr><td>FC $\#units = 15$, Linear</td><td>—</td></tr>
<tr><td>softmax(dim=15)</td><td>softmax(dim=60×60)</td></tr>
</table>

the kernel size and the number of feature maps for conv layers and the number of units for FC layers change similar to Seq-MT model on those datasets.

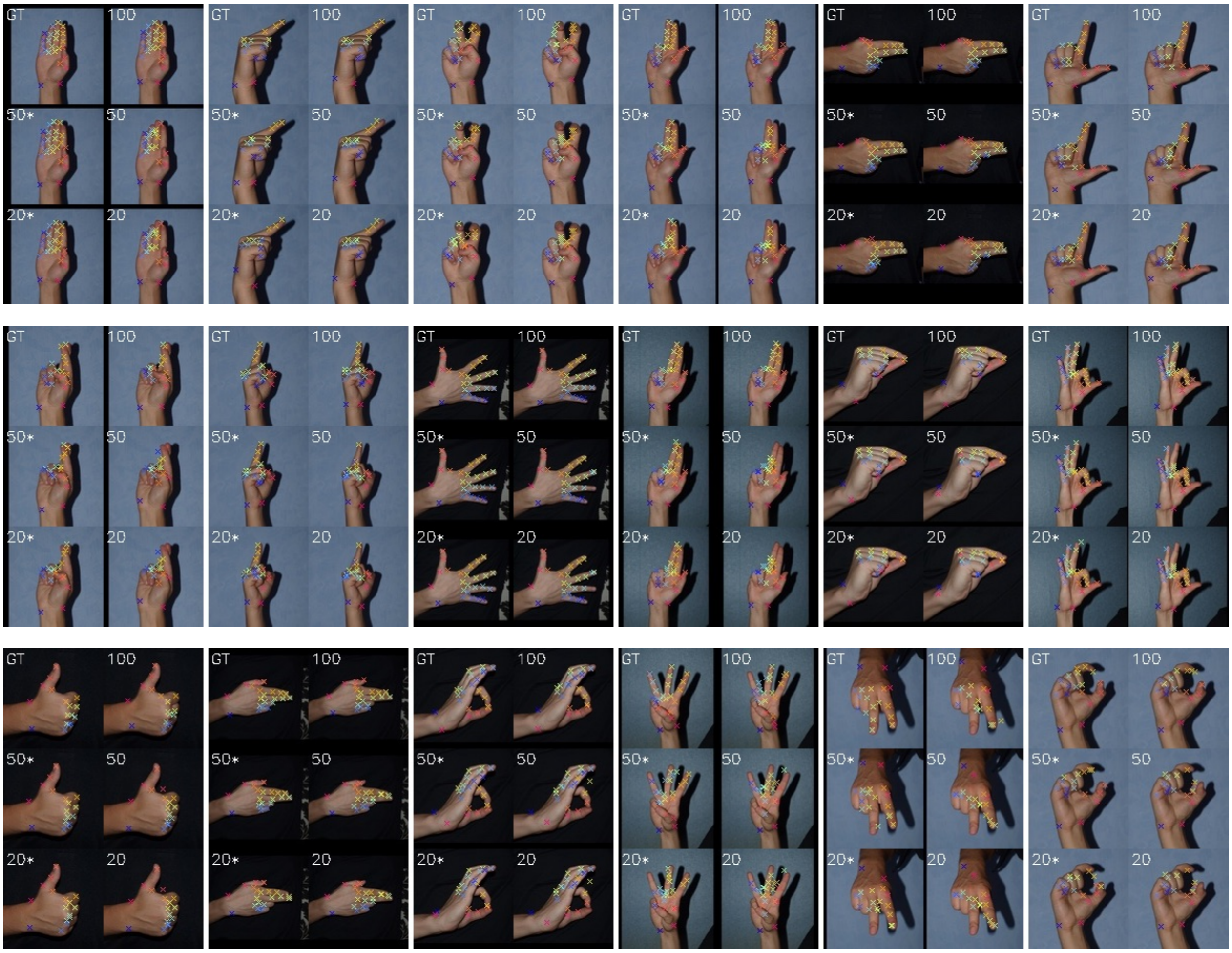

Figure S2: Extra examples of our model predictions on the HGR1 [16, 23] test set. GT represents ground-trust annotations, while numbers 100, 50, and 20 indicate the percentage of the training set with labeled landmarks. Results are computed with Seq-MT (L+ELT+A) model (denoted *) and Seq-MT (L). Examples illustrate improvement of the landmark prediction by using the class label and the ELT cost in addition to the labeled landmarks. The last three examples on the bottom row show examples with high errors. Best viewed in color with zoom.

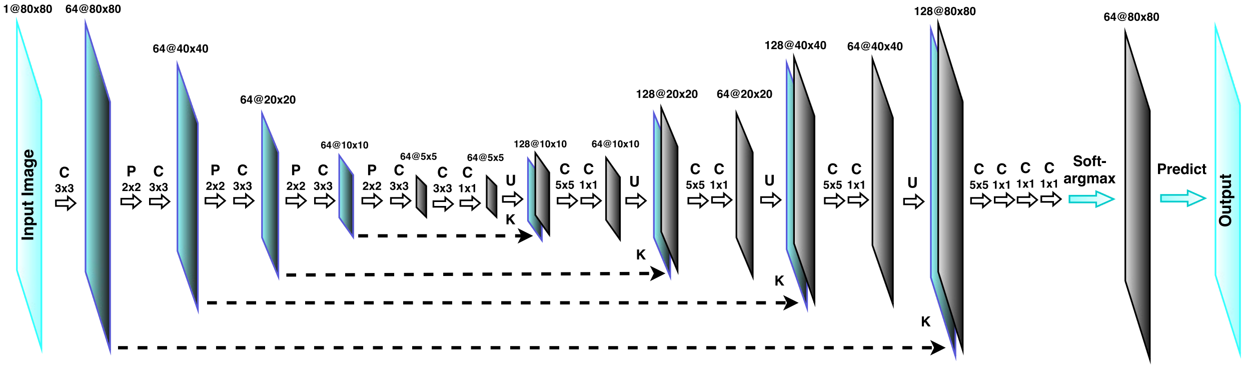

Figure S3: The ReCombinator Networks (RCN) [13] architecture used for experiments on 300W dataset. P indicates a pooling layer. All pooling layers have stride of 2. C indicates a convolutional layer. The number written below C indicates the convolution kernel size. All convolutions have stride of 1. U indicates an upsampling layer, where each feature map is upsampled to the next (bigger) feature map resolution. K indicates concatenation, where the upsampled features are concatenated with features of the same resolution before a pooling is applied to them. The dashed arrows indicate the feature maps are carried forward for concatenation. The solid arrows following each other, e.g. P, C, indicate the order of independent operations that are applied. The number written above feature maps in $n@w \times h$ format indicate number of feature maps n and the width w and height h of the feature maps. On AFLW, we use 70 feature maps per layer (instead of 64) and we get two levels coarser to get to 1×1 resolution (instead of 5×5). On both datasets we shoud $\beta = 100$ for soft-argmax layer.

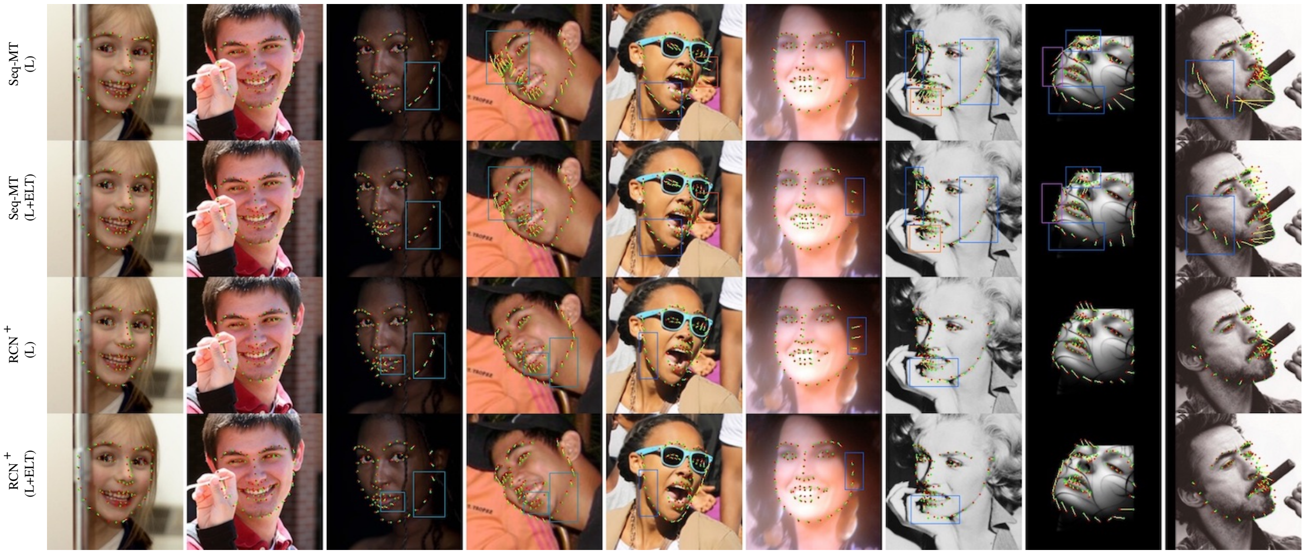

Figure S4: Extra examples of our model predictions on 300W [27] test-set. The first two columns depict examples where all models get accurate predictions, The next 5 columns illustrate the improved accuracy obtained by using ELT loss in two different architectures (Seq-MT and RCN). The last two columns show difficult examples where error is high. The rectangles indicate the regions that landmarks are mostly affected. The green and red dots show ground truth (GT) and model predictions (MP), respectively. The yellow lines show the error by connecting GT and MP. Note that the ELT loss improves predictions in both architectures. Best viewed in color with zoom.

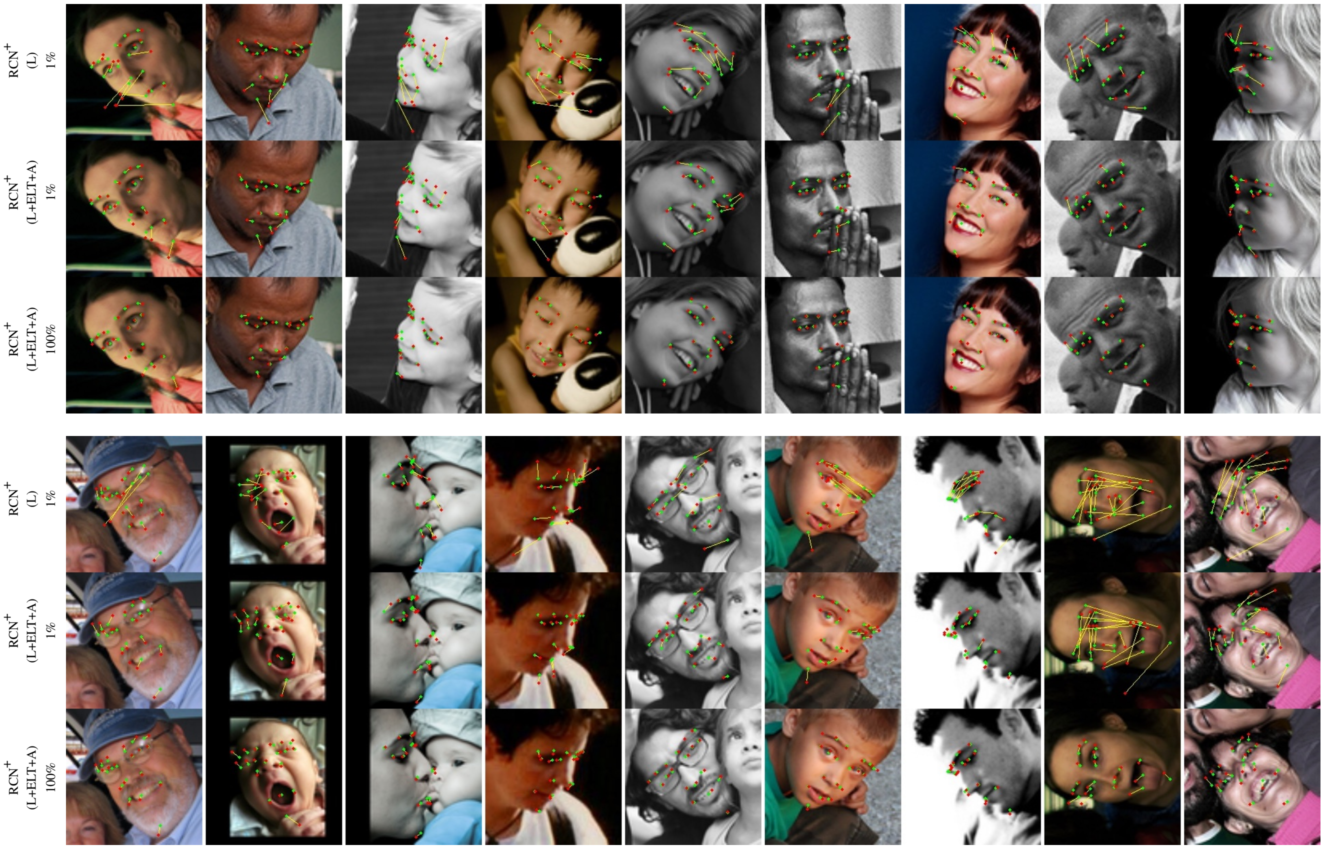

Figure S5: Extra examples of our model predictions on the AFLW test set. Comparing the first and second rows shows the improvement obtained by using ELT+A with only 1% of labelled landmarks. Note the model trained using ELT+A preserves better the distribution over the landmarks. The last two columns in the bottom row show samples with high error on small percentage of labelled landmaks, which is due to extreme rotation. The bottom row shows the prediction using L+ELT+A on the entire set of labelled landmarks, which gets the best results. The green and red dots show ground truth (GT) and model predictions (MP), respectively. The yellow lines show the error by connecting GT and MP. Best viewed in color with zoom.

Table S11: Architecture details of Seq-MT model used for Shapes and Blocks datasets. Each conv layer has three values as $w \times h \times n$ indicating width (w), height (h) of kernel and the number of feature maps (n) of the convolutional layer. SAME indicates the input map is padded with zeros such that input and output maps have the same resolution.

Shapes Dataset	**Blocks Dataset**
Model HP: $\lambda = 0, \alpha = 0, \gamma = 0, \beta = 1$, ADAM	Model HP: $\lambda = 1, \alpha = 1, \beta = 1$, ADAM
Landmark Localization Network	Landmark Localization Network
Input = $60 \times 60 \times 1$	Input = $60 \times 60 \times 1$
Conv $7 \times 7 \times 16$, ReLU, stride 1, SAME	Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME
Conv $7 \times 7 \times 16$, ReLU, stride 1, SAME	Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME
Conv $7 \times 7 \times 16$, ReLU, stride 1, SAME	Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME
Conv $7 \times 7 \times 16$, ReLU, stride 1, SAME	Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME
Conv $7 \times 7 \times 16$, ReLU, stride 1, SAME	Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME
Conv $7 \times 7 \times 16$, ReLU, stride 1, SAME	Conv $9 \times 9 \times 8$, ReLU, stride 1, SAME
Conv $1 \times 1 \times 16$, ReLU, stride 1, SAME	Conv $1 \times 1 \times 8$, ReLU, stride 1, SAME
Conv $1 \times 1 \times 2$, ReLU, stride 1, SAME	Conv $1 \times 1 \times 5$, ReLU, stride 1, SAME
soft-argmax(num_channels=2)	soft-argmax(num_channels=5)
Classification Network	Classification Network
FC $\#units = 40$, ReLU	FC $\#units = 256$, ReLU, dropout-prob=.25
FC $\#units = 2$, Linear	FC $\#units = 256$, ReLU, dropout-prob=.25
	FC $\#units = 15$, Linear
softmax(dim=2)	softmax(dim=15)

Table S12: Architecture details of Seq-MT model used for Hands and Multi-PIE datasets.

Hands Dataset	**Multi-PIE Dataset**	
Model HP: $\lambda = 0.5, \alpha = 0.3, \gamma = 10^{-5}, \beta = 0.001$, ADAM	Model HP: $\lambda = 2, \alpha = 0.3, \gamma = 10^{-5}, \beta = 0.001$, ADAM	
Preprocessing: scale and translation [-10%, 10%] of face bounding box, rotation [-20, 20] applied randomly to every epoch.		
Landmark Localization Network	Landmark Localization Network	
Input = $64 \times 64 \times 1$	Input = $64 \times 64 \times 1$	
Conv $9 \times 9 \times 64$, ReLU, stride 1, SAME	Conv $9 \times 9 \times 64$, ReLU, stride 1, SAME	
Conv $9 \times 9 \times 64$, ReLU, stride 1, SAME	Conv $9 \times 9 \times 64$, ReLU, stride 1, SAME	
Conv $9 \times 9 \times 64$, ReLU, stride 1, SAME	Conv $9 \times 9 \times 64$, ReLU, stride 1, SAME	
Conv $9 \times 9 \times 64$, ReLU, stride 1, SAME	Conv $9 \times 9 \times 64$, ReLU, stride 1, SAME	
Conv $9 \times 9 \times 64$, ReLU, stride 1, SAME	Conv $9 \times 9 \times 64$, ReLU, stride 1, SAME	
Conv $9 \times 9 \times 25$, ReLU, stride 1, SAME	Conv $9 \times 9 \times 68$, ReLU, stride 1, SAME	
soft-argmax(num_channels=25)	soft-argmax(num_channels=68)	
Classification Network	Emotion Classification Branch	Camera Classification Branch
FC $\#units = 256$, ReLU, dropout-prob=.5	FC $\#units = 256$, ReLU, dropout-prob=.25	FC $\#units = 256$, ReLU, dropout-prob=.25
FC $\#units = 256$, ReLU, dropout-prob=.5	FC $\#units = 256$, ReLU, dropout-prob=.25	FC $\#units = 256$, ReLU, dropout-prob=.25
FC $\#units = 27$, Linear	FC $\#units = 6$, Linear	FC $\#units = 5$, Linear
softmax(dim=27)	softmax(dim=6)	softmax(dim=5)

Table S13: Architecture details of Seq-MT model used for 300W datasets.

300W Dataset
Model HP: $\lambda = 2.0, \alpha = 2.0, \gamma = 10^{-5}, \beta = 0.01$, ADAM
Preprocessing: scale and translation [-10%, 10%] of face bounding box, rotation [-30, 30] applied randomly to every epoch.
Landmark Localization Network
Input = $64 \times 64 \times 1$
Conv $9 \times 9 \times 32$, ReLU, stride 1, SAME
Conv $9 \times 9 \times 32$, ReLU, stride 1, SAME
Conv $9 \times 9 \times 32$, ReLU, stride 1, SAME
Conv $9 \times 9 \times 32$, ReLU, stride 1, SAME
Conv $9 \times 9 \times 32$, ReLU, stride 1, SAME
Conv $9 \times 9 \times 32$, ReLU, stride 1, SAME
Conv $9 \times 9 \times 32$, ReLU, stride 1, SAME
Conv $9 \times 9 \times 68$, ReLU, stride 1, SAME
soft-argmax(num_channels=68)