# Supervision by Registration and Triangulation for Landmark Detection

Xuanyi Dong, Yi Yang, Shih-En Wei, Xinshuo Weng, Yaser Sheikh, Shoou-I Yu

**Abstract**—We present Supervision by Registration and Triangulation (SRT), an unsupervised approach that utilizes unlabeled multi-view video to improve the accuracy and precision of landmark detectors. Being able to utilize unlabeled data enables our detectors to learn from massive amounts of unlabeled data freely available and not be limited by the quality and quantity of manual human annotations. To utilize unlabeled data, there are two key observations: (I) the detections of the same landmark in adjacent frames should be coherent with registration, i.e., optical flow. (II) the detections of the same landmark in multiple synchronized and geometrically calibrated views should correspond to a single 3D point, i.e., multi-view consistency. Registration and multi-view consistency are sources of supervision that do not require manual labeling, thus it can be leveraged to augment existing training data during detector training. End-to-end training is made possible by differentiable registration and 3D triangulation modules. Experiments with 11 datasets and a newly proposed metric to measure precision demonstrate accuracy and precision improvements in landmark detection on both images and video.

**Index Terms**—Landmark Detection, Optical Flow, Triangulation, Deep Learning

## 1 Introduction

Accurate and precise landmark detection is an important component to high quality performance of many computer vision and computer graphics tasks, such as face tracking [1], [2], [3], face reenactment [4], and body tracking [5], [6]. In many of these tasks, the landmarks are used to either provide good initialization for subsequent processing [1], [2], [5], [7], or as loss terms which anchor the tracked face/body in an energy minimization or deep learning scenario [3], [4], [6]. Therefore, any errors such as inaccurate landmarks or unstable landmarks could propagate through the entire pipeline and have adverse effects on the final results. For example, in face or body mesh tracking, 2D landmarks are used as anchors to deform a template mesh to match the current observations, so inaccuracies and instabilities in landmark detections could propagate to the tracked mesh and generate perceptually jarring results [8].

While significant amount of work has been done on image-based landmark detection [9], [10], [11], [12], [13], [14], landmark detection accuracy and stability is far from optimal. In the ideal case, independent frame-by-frame detections on a video sequence should be as accurate as if a marker was physically attached to the face or body. However, high-frequency jitter is still observed when visualizing the frame-by-frame detections from state-of-the-art models [8], suggesting that detector performance can be further improved.

In order to improve detector performance, we first separate detector performance in two aspects, accuracy and precision, to facilitate analysis. An *accurate* detector predicts landmarks that are consistent to manual annotations. A *precise* detector localizes a landmark at the exact same semantic location across different input images. Note that an accurate detector is usually also precise, but a precise detector does not necessarily mean it is accurate, i.e., the detections can be consistent across different input images but still far from human annotations. One extreme example is the detector can always predict $(0, 0)$, which is semantically very consistent (very precise) but also very inaccurate. On the other hand, the accuracy metric heavily depends on the quality of human annotations which have inherent limits on consistency as shown in Figure 1, and noisy annotations in the test set could make the detector accuracy look worse than it actually is as shown in [8]. This is not the case for precision, where our proposed approach (see Section 4.2) can compute precision without relying on annotations, thus it is no longer affected by inconsistencies in annotations. In sum, both metrics have its merits and limitations, so in this paper, we measure both the accuracy and precision of the detector to get the most comprehensive story.

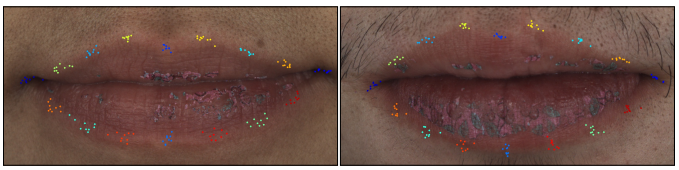


Fig. 1. **Annotations are inconsistent.** We show annotations of nine annotators on two images of the mouth. Each color indicates a different landmark. Note the inconsistencies of annotations exist even on the more discriminative landmarks such as the corner of the mouth.

Based on the aforementioned metrics, we analyze the causes of unstable predictions, which could be due to: (I) insufficient training samples, and (II) imprecise annotations.


- *Xuanyi Dong and Yi Yang are with Centre for Artificial Intelligence, University of Technology Sydney, NSW, Australia. (e-mail: xuanyi.dong@student.uts.edu.au; yi.yang@uts.edu.au)*
- *Shoou-I Yu, Shih-En Wei, and Yaser Sheikh are with Facebook Reality Labs, Pittsburgh, USA. (e-mail: shoou-i.yu@fb.com; shih-en.wei@fb.com; yaser.sheikh@fb.com)*
- *Xinshuo Weng is with the Robotics Institute, Carnegie Mellon University, Pittsburgh, USA. (e-mail: xinshuow@cs.cmu.edu)*

In Figure 5, we analyzed the effect of varying the number of training samples and error of annotations on a synthetic dataset, which can provide ground-truth with zero annotation error. Results show that both accuracy and precision increases as we increase the number of training samples, and as we add noise to the annotations used for training, both accuracy and precision drop significantly. We observe that a model which was trained on annotation data with zero error achieves better performance than a model trained on twice the amount of noisy annotations. This suggests that achieving high quality landmark detection might require very large amounts of imprecise human annotations, which is very labor intensive to collect.

Instead of completely relying on human annotations and being limited by their quality and quantity, we present *Supervision by Registration and Triangulation* (SRT), a method which augments the training loss function with supervision automatically extracted from *unlabeled* multi-view videos. SRT consists of Supervision-by-Registration (SBR) and Supervision-by-Triangulation (SBT), which we detail in the following paragraphs.

For SBR, the consistency of (I) the detections of the same landmark in adjacent video frames and (II) registration, i.e., optical flow, is a source of supervision. More specifically, a detected landmark at $\text{frame}_{t-1}$ followed by optical flow tracking between $\text{frame}_{t-1}$ and $\text{frame}_t$ should coincide with the location of the detection at $\text{frame}_t$. Therefore, if the detections are incoherent with the optical flow, the mismatch is a supervisory signal enforcing the detector to be temporally consistent across frames, thus enabling a SBR-trained detector to better locate the correct location of a landmark which may be hard to annotate precisely.

For SBT, the consistency of (I) the detections of the same landmark in different synchronized plus geometrically calibrated views and (II) the 3D triangulation constraint, is a source of supervision. More specifically, a detected landmark in different views should coincide with the reprojected landmarks calculated via 3D triangulation. Therefore, if the detections are incoherent with the reprojected landmarks, the mismatch is a supervisory signal enforcing the detector to be spatially consistent across views.

The overview of our method is shown in Figure 2 and Figure 3. Our end-to-end trainable model consists of three modules: a generic detector built on convolutional neural networks [12], [13], a differentiable optical flow (OF) module, and a differentiable 3D triangulation (3DT) module. During the forward pass, the OF module takes the landmark detections from the past frame and estimates their locations in the current frame. The tracked landmarks are then compared with the detections on the current frame. The registration loss is defined as the offset between them. The 3DT module takes the landmark detections from different views and estimates the 3D location. The reprojected landmarks from the 3D location are then compared with the detections in each view. The multi-view loss is defined as the discrepancy between them. In the backward pass, the gradients from the registration loss and the multi-view loss are back-propagated through the OF and 3DT modules to encourage spatial-temporal coherency in the detector. The final output of our method is an enhanced image-based landmark detector, which has leveraged large amounts of unlabeled synchronized and geometrically calibrated multi-view video to achieve higher accuracy and precision in both images and videos, more stable predictions in videos, and more consistent predictions in different views.

Note that our approach is fundamentally different from post-processing such as temporal filtering, which often sacrifices precision for stability. Our method directly incorporates the supervision of registration and multi-view coherency during model *training*, thus producing detectors that are inherently more stable. Therefore, during testing time, neither post-processing, optical flow tracking, nor recurrent units are required. Also note that SRT is not regularization, which limits the freedom of model parameters to prevent over-fitting. Instead, SRT brings more supervisory signals from registration and triangulation to enhance the accuracy and precision of the detector.

In order to evaluate our claims, we perform a series of experiments to validate our method. The questions we try to answer are as follows:
(I) Do both accuracy and precision improve through SRT?
(II) Which one is more useful, SBR or SBT? And do they complement each other?
(III) How do the quality and quantity of unlabeled videos affect the accuracy and precision of the detector?
(IV) When do SBR and SBT not work?

Experiments on both regression and heatmap-based detectors show that SBR and SBT are both helpful and can complement each other in improving accuracy and precision for landmark detection. The improvement is most significant when the distribution of the unlabeled data is similar to the labeled data, and inversely, when the distributions differ greatly, SBR and SBT may actually harm performance. Also, there is a low chance (0.46% in our optical flow experiments in Section 4.3) that incorrect detections are still consistent with optical flow or multi-view constraints, which could lead to the detector learning from incorrect supervision and hurting performance.

In sum, SRT has the following benefits:
(I) SRT can enhance the accuracy and precision of a generic landmark detector on both images and multi-view video without requiring additional labeled data.
(II) Since the supervisory signal of SRT does not come from annotations, SRT can leverage a very large amount of unlabeled synchronized and geometrically calibrated multi-view video to enhance the detector, thus SRT is no longer limited by the quality and quantity of manual human annotations.
(III) SRT can be trained end-to-end.
(IV) SRT is only applied during *training* time, thus the prediction speed at test time is not affected.

## 2 Related Work

Landmark detection algorithms are mainly applied to three modalities: images, video, and multi-view. In images, the detector can only rely on the static image to detect landmarks, whereas in video the detector has additional temporal information to utilize. In multi-view systems, additional geometric information is available to further boost the performance of a detector. We will briefly compare our method with other detection algorithms in these three modalities.

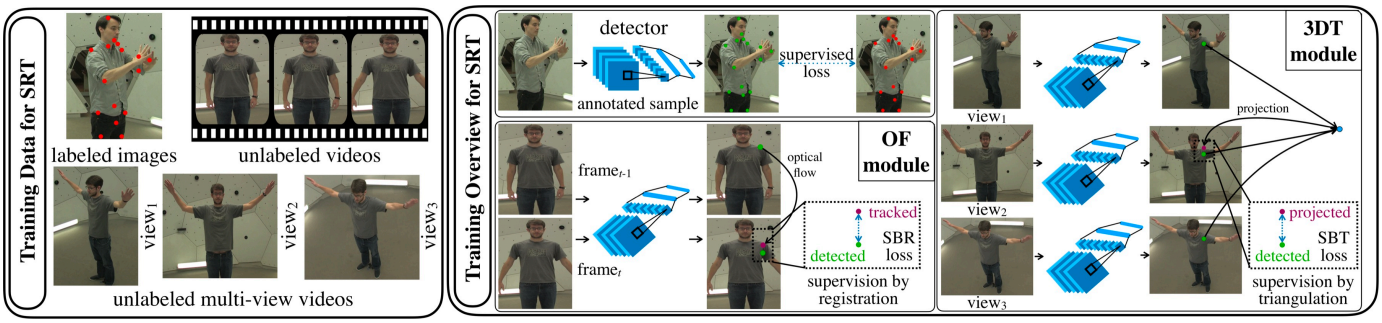



Fig. 2. The **Supervision by Registration and Triangulation (SRT) framework** takes labeled images and unlabeled synchronized and geometrically calibrated multi-view video as input to train an image-based landmark detector which is more precise on images/video, more stable on video, and also more consistent in multi-view scenarios. OF and 3DT stands for Optical Flow and 3D Triangulation respectively.

### 2.1 Image-based Landmark Detection

Despite some early works that learn to regress the coordinates of each landmark from hand-crafted features [11], recent landmark detection methods take advantage of end-to-end training from the deep convolutional neural network (CNN) model [12], [13]. A typical method was to append an linear regression layer to predict the coordinates right after the CNN features and train the network in an end-to-end fashion. To improve performance, one option is to cascade multiple CNN models to progressively refine the predictions [15], [16], [17]. For example, Yu et al. [18] proposed a deformation network to incorporate geometric constraints in the CNN framework. Zhu et al. [15] leveraged cascaded regressors to handle extreme head poses and rich shape deformation. Lv et al. [9] presented a two-stage architecture to explicitly deal with the poor initialization problem.

Another category of landmark detection methods learn to predict a heatmap for each landmark [12], [19], [20], [21], [22]. Specifically, these methods define a Gaussian map at each ground-truth landmark coordinate, and the network aims to output that Gaussian map. Wei et al. [12] and Newell et al. [13] took the location with the highest response of the heatmap as the coordinate of the corresponding landmarks. Li et al. [19] enhanced the landmark detection through multi-task learning. Bulat et al. [20] proposed a robust network structure to utilize the advanced residual hourglass architectures.

Our proposed SRT is orthogonal to these image-based algorithms in that SRT can enhance both regression and heatmap-based detectors as we demonstrate in Section 4.

### 2.2 Video-based Landmark Detection

Though image-based landmark detectors can achieve very good performance on images, sequentially running these detectors on each frame of multi-view videos in a tracking-by-detection fashion usually leads to jittering, unstable, and inconsistent detections. One way to decrease jittering is to just initialize the tracker with the detection once then perform temporal tracking [23], but this suffers from tracker drift. Once the tracker has failed in the current frame, it is difficult to make the correct prediction in the subsequent frames. Therefore, hybrid methods [24], [25], [26], [27] jointly utilize tracking-by-detection and temporal information in a single framework to predict more stable landmarks. Peng et al. [26] and Liu et al. [25] utilized recurrent neural networks to encode the temporal information across consecutive frames. Khan et al. [24] utilized global variable consensus optimization to jointly optimize detection and tracking in consecutive frames. Unfortunately, these methods require per-frame annotations, which are not only resource-intensive to acquire, but also difficult to annotate consistently across frames, even for temporally adjacent frames. Even thought SRT shares the same high-level idea of these algorithms by leveraging temporal coherency, but SRT does not require any video-level annotation, and is therefore not limited by the availability and precision of human annotations.

Other approaches utilize temporal information in video to construct person-specific models [28]. Most of these methods usually leverage offline-trained static appearance models, i.e., the detector, which is used to generate initial landmark prediction, is not updated based on the tracking result in their algorithms, whereas SBR in our SRT dynamically refines the detector based on OF tracking results. Self-training [29] can also be utilized for creating person-specific models, and was shown to be effective in pose estimation [30], [31]. However, unlike our method which can be trained end-to-end, [30], [31] did alternating bootstrapping to progressively improve the detectors. These methods make hard decisions on whether a pseudo-labeled sample should be added to the training set or not, and may also suffer from inaccurate gradient updates [8].

### 2.3 Multi-view Landmark Detection

Leveraging geometrically calibrated and synchronized multi-view images enable us to utilize epipolar constraints to enhance the detector. A straightforward solution is bootstrapping [32]. Starting from an initial weak detector, triangulation and reprojection is used to generate robust pseudo-labels on unlabeled images, which is then used to enhance the initial detector. Other researchers took this approach one step further by incorporating the multi-view loss into their objective and train the model in an end-to-end fashion [33], [34], [35]. Suwajanakorn et al. [33] applied a multi-view consistency loss between every view pair in their framework. The authors of [34], [35] leveraged the epipolar constraint to reduce the mismatch between predictions in two different views. Rhodin et al. [36] trained the system to predict the same pose in all views. Amberg et al. [37] proposed to align

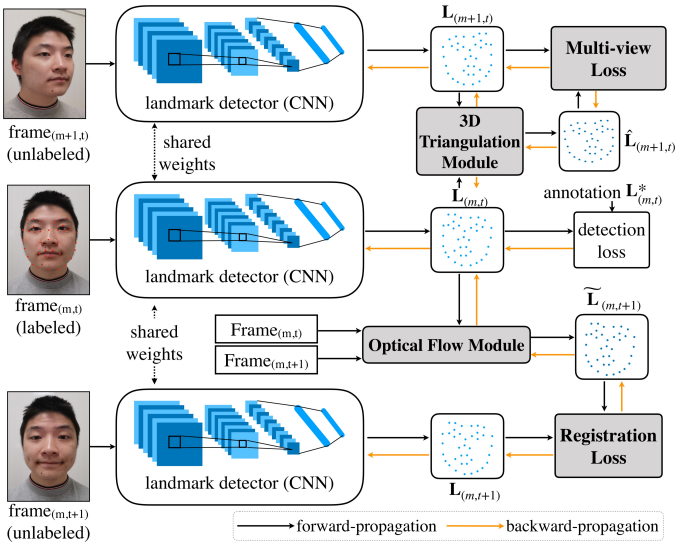



Fig. 3. The **training** of SRT with three complementary losses. The key idea is that the supervision from registration and triangulation can directly back-propagate through the optical flow (OF) and 3D triangulation (3DT) modules respectively, thus enabling the detector before the OF and 3DT modules to receive gradients which encourage temporal and multi-view coherency across frames.

the raw pixels of corresponding points in different views for 3D face reconstruction. Pavllo et al. [38] proposed a semi-supervised training strategy by minimizing the bone length via a 3D model. SRT shares the same high-level idea of these methods by utilizing multi-view coherency, and one advantage of SRT is we are able to utilize multiple views, instead of just two views ( [34], [35]), jointly in one training iteration, thus being able to leverage information from multiple views at once.

## 3 Methodology

SRT consists of three complementary parts, the general landmark detector, the optical flow (OF) module, and the 3D triangulation (3DT) module, as shown in Figure 3. Each part has a corresponding loss function. The detection loss utilizes appearance from a single image and label information to learn a better landmark detector. The registration and multi-view losses leverage the differentiable OF and 3DT modules respectively to enforce the predictions in neighboring frames and different views to be consistent. The key components to SRT are the OF and 3DT modules, which we describe in detail in the following sections.

### 3.1 Differentiable Optical Flow Module

For SBR, one essentially needs a differentiable OF module that takes as input a location on the image and outputs the OF at that location. We present two possible directions to implement this module.

The first direction is to decompose an existing OF algorithm into a series of differentiable functions, thus enabling us to directly incorporate the OF algorithm into deep network training. In Section 3.1.1, we use the Lucas Kanade optical flow algorithm [39] as an example. However, this method tends to be too computationally inefficient in practice. Even though the theoretical computation of optical flow is negligible compared to the complexity of evaluating a CNN, the actual speed of computing OF is significantly lower than evaluating a CNN. This is mainly due to existing GPU libraries being highly optimized for common CNN modules such as convolutions, but not optimized for custom modules such as differentiable Lucas Kanade tracking. Therefore, we also propose another direction which can efficiently approximate OF.

The second direction is to approximate OF at sub-pixel locations through bilinear interpolation on a pre-computed flow field, which is still a fully differentiable process. This enables us to remove OF compute time out of network forward and backward passes, thus providing significant boosts in training speeds (please see analysis in supplementary material). It also enables us to utilize more sophisticated and potentially non-differentiable OF algorithms.

We now detail each direction in the following sections and explain the key notations in Table 1.

#### 3.1.1 Differentiable Lucas Kanade OF

Motivated by [40], we decompose the Lucas Kanade OF algorithm into a series of differentiable steps, such that gradients can back-propagate through the optical flow module.

Given the feature $\boldsymbol{F}_{(m,t-1)}$[1] from frame$_{(m,t-1)}$ in view$_m$ (the $m$-th view) at time$_{t-1}$ (the $t$-1-th timestamp) and feature $\boldsymbol{F}_{(m,t)}$ from frame$_{(m,t)}$, we estimate the motion for a small patch near $\boldsymbol{x}_{(m,t-1)} = [x, y]^T$ from frame$_{(m,t-1)}$. The motion model is represented by the displacement warp function $W(\boldsymbol{x}; \boldsymbol{p})$. A displacement warp contains two parameters $\boldsymbol{p} = [p_1, p_2]^T$, and can be formulated as $W(\boldsymbol{x}; \boldsymbol{p}) = [x + p_1, y + p_2]^T$. We leverage the inverse compositional algorithm [23] for our OF operation. It finds the motion parameter $\boldsymbol{p}$ by minimizing

$$\sum_{\boldsymbol{x} \in \Omega} \alpha_{\boldsymbol{x}} \parallel \boldsymbol{F}_{(m,t-1)}(W(\boldsymbol{x}; \Delta\boldsymbol{p})) - \boldsymbol{F}_{(m,t)}(W(\boldsymbol{x}; \boldsymbol{p})) \parallel^2, \quad (1)$$

with respect to $\Delta\boldsymbol{p}$. Here, $\Omega$ is a set of locations in a patch centered at $\boldsymbol{x}_{(m,t-1)}$, and $\alpha_{\boldsymbol{x}} = \exp(-\frac{||\boldsymbol{x}-\boldsymbol{x}_{(m,t-1)}||_2^2}{2\sigma^2})$ is the weight value for $\boldsymbol{x}$ determined by the distance from $\boldsymbol{x}_{(m,t-1)}$ to down-weight pixels further away from the center of the patch. After obtaining $\Delta\boldsymbol{p}$, we update $\boldsymbol{p}$ as follows:

$$W(\boldsymbol{x}; \boldsymbol{p}) \leftarrow W(W(\boldsymbol{x}; \Delta\boldsymbol{p})^{-1}; \boldsymbol{p}) = \begin{bmatrix} x + p_1 - \Delta p_1 \\ y + p_2 - \Delta p_2 \end{bmatrix}. \quad (2)$$

$\boldsymbol{p}$ is an initial motion parameter ($\boldsymbol{p} = [0, 0]$ in our case), which will be iteratively updated by Eq. (2) until convergence.

For each iteration, $\Delta\boldsymbol{p}$ is computed through minimizing the first order Taylor expansion of Eq. (1):

$$\sum_{\boldsymbol{x} \in \Omega} \alpha_{\boldsymbol{x}} \parallel \boldsymbol{F}_{(m,t-1)}(W(\boldsymbol{x}; \boldsymbol{0})) + \nabla \boldsymbol{F}_{(m,t-1)} \frac{\partial W}{\partial \boldsymbol{p}} \Delta\boldsymbol{p} - \boldsymbol{F}_{(m,t)}(W(\boldsymbol{x}; \boldsymbol{p})) \parallel^2. \quad (3)$$

The $\Delta\boldsymbol{p}$ which minimizes Eq. (3) is:

$$\Delta\boldsymbol{p} = \boldsymbol{H}^{-1} \sum_{\boldsymbol{x} \in \Omega} J(\boldsymbol{x})^T \alpha_{\boldsymbol{x}} \times (\boldsymbol{F}_{(m,t)}(W(\boldsymbol{x}; \boldsymbol{p})) - \boldsymbol{F}_{(m,t-1)}(W(\boldsymbol{x}; \boldsymbol{0}))), \quad (4)$$

1. The features can be RGB images or outputs of convolutional layers.

**Algorithm 1** The differentiable Lucas Kanade OF operation.

**Input:** $\boldsymbol{F}_{(m,t-1)}$, $\boldsymbol{F}_{(m,t)}$, and $\boldsymbol{p} = [0, 0]$
**Input:** the $k$-th landmark location $\boldsymbol{x}$, i.e., $\boldsymbol{L}_{(m,t-1,k)}$
1. Extract template feature from $\boldsymbol{F}_{(m,t-1)}$ centered at $\boldsymbol{x}$
2. Calculate the gradient of the template feature
3. Compute the Jacobian and Hessian matrices, $\boldsymbol{J}$ and $\boldsymbol{H}$
**for** iter=1; iter $\leq$ max; iter++ **do**
4. Extract target feature from $\boldsymbol{F}_{(m,t)}$ centered at $\boldsymbol{x} + \boldsymbol{p}$
5. Compute difference of template and target features
6. Compute $\Delta\boldsymbol{p}$ using Eq. (4)
7. Update the motion model $\boldsymbol{p}$ using Eq. (2)
**end for**
**Output:** the $k$-th landmark location at frame$_{(m,t)}$ : $\boldsymbol{x} + \boldsymbol{p}$

where $\boldsymbol{H}$=$\boldsymbol{J}^T\boldsymbol{A}\boldsymbol{J} \in \mathcal{R}^{2\times2}$ is the Hessian matrix. $\boldsymbol{J} \in \mathcal{R}^{C|\Omega|\times2}$ is the vertical stacking of $J(\boldsymbol{x}) \in \mathcal{R}^{C\times2}$, $\boldsymbol{x} \in \Omega$, which is the Jacobian matrix of $\boldsymbol{F}_{(m,t-1)}(W(\boldsymbol{x}; \boldsymbol{0}))$. $C$ is the number of channels of $\boldsymbol{F}$. $\boldsymbol{A}$ is a diagonal matrix, where elements in the main diagonal are the $\alpha_{\boldsymbol{x}}$'s corresponding to the $\boldsymbol{x}$'s used to compute each row of $\boldsymbol{J}$. $\boldsymbol{H}$ and $\boldsymbol{J}$ are constant over iterations and can thus be pre-computed.

We describe the detailed steps of the Lucas Kanade OF operation in Alg. 1. We define the OF operation as $\widetilde{\boldsymbol{L}}_{(m,t,k)} = G_{OF}(\boldsymbol{F}_{(m,t-1)}, \boldsymbol{F}_{(m,t)}, \boldsymbol{L}_{(m,t-1,k)})$. This function takes the detected location of the $k$-th landmark in frame$_{(m,t-1)}$: $\boldsymbol{L}_{(m,t-1,k)} \in \mathcal{R}^2$, as input computes the landmark location $\widetilde{\boldsymbol{L}}_{(m,t,k)}$ for the next (future) frame. Since, all steps in the OF operation are differentiable, the gradient can back-propagate to the facial landmark detections and the feature maps.

We add a very small value to the diagonal elements of $\boldsymbol{H}$. This ensures that $\boldsymbol{H}$ is invertible. Also, in order to crop a patch at a sub-pixel location $\boldsymbol{x}$, we use the spatial transformer network [41] to calculate the bilinear interpolated values of the feature maps.

**Loss calculation**. SBR supervision comes from the discrepancy between the predicted coordinates $\boldsymbol{L}_{(m,t,k)}$ and the estimated coordinates $\widetilde{\boldsymbol{L}}_{(m,t,k)}$. The loss function that calculates this discrepancy can vary for regression and heatmap-based detectors. We will introduce our choices in Section 3.3.

#### 3.1.2 Bilinear Interpolation Approximation of OF

Another method to compute OF is to approximate OF at sub-pixel locations through bilinear interpolation on a pre-computed flow field, which is still a fully differentiable process. An issue is that such approximation with bilinear interpolation might lead to drop in optical flow accuracy. Fortunately, we empirically show that the OF computed from bilinear interpolation is very accurate (see details in Section 4.3.) Given that the OF error due to interpolation is minimal, this enables us to leverage many more sophisticated OF algorithms.

### 3.2 Differentiable 3D Triangulation Module

To incorporate multi-view consistency, we design a 3DT module through which we can perform back-propagation. There are three steps: triangulation, reprojection, and loss calculation. Given the predicted coordinate of a landmark from all synchronized and calibrated views, we first estimate the 3D coordinates through triangulation. Then, the 3D point is reprojected back into each camera view. Finally, the location difference between the reprojected and the detected points are used to compute the loss. These steps are repeated for all landmarks. Details for each step are as follows.

TABLE 1
Explanation of notations in this manuscript.

| Notation | Definition |
|---|---|
| $m$ | the index for views |
| $t$ | the index for time frames |
| $k$ | the index for landmarks |
| $\boldsymbol{F}_{(m,t)}$ | the feature tensor from the $t$-th frame in the $m$-th view |
| $\boldsymbol{x}$ | a 2D coordinate |
| $\boldsymbol{p}$ | a 2D displacement |
| $W(\boldsymbol{x}, \boldsymbol{p})$ | translate the coordinate $\boldsymbol{x}$ with $\boldsymbol{p}$ |
| $\Delta\boldsymbol{p}$ | a 2D vector to update $\boldsymbol{p}$ |
| $\boldsymbol{H}$ | the Hessian matrix |
| $K$ | the number of landmarks |
| $\boldsymbol{L}_{(m,t,k)}$ | the predicted $k$-th landmark at frame$_t$ in view$_m$ |
| $\widetilde{\boldsymbol{L}}_{(m,t,k)}$ | the OF-tracked $k$-th landmark at frame$_t$ in view$_m$ |
| $\hat{\boldsymbol{L}}_{(m,t,k)}$ | the 3DT-computed $k$-th landmark at frame$_t$ in view$_m$ |
| $\boldsymbol{L}^*_{(m,t,k)}$ | ground truth label of $k$-th landmark at frame$_t$ in view$_m$ |
| $\boldsymbol{M}_{(m,t,k)}$ | the $k$-th predicted heatmap at frame$_t$ in view$_m$ |
| $\mathbb{M}_m \in \mathcal{R}^{3\times4}$ | the camera transformation matrix for the $m$-th view |

**3D Triangulation**. We solve for the 3D location of a landmark through Direct Linear Transformation [42]. Given the camera transformation matrices $\mathbb{M}_m \in \mathcal{R}^{3\times4}$ for all $M$ views, and detections in each view $\boldsymbol{L}_{(m,t,k)} \in \mathcal{R}^2$, $1 \leq m \leq M$, where $m$ indicates the index of views, we can calculate the 3D landmark $\boldsymbol{L}^{\text{3D}}_{(t,k)} \in \mathcal{R}^3$ as follows:

$$\boldsymbol{u}_m = \mathbb{M}_m[0,:] - \mathbb{M}_m[2,:] \cdot \boldsymbol{L}_{(m,t,k)}[0] \in \mathcal{R}^4, \tag{5}$$

$$\boldsymbol{v}_m = \mathbb{M}_m[1,:] - \mathbb{M}_m[2,:] \cdot \boldsymbol{L}_{(m,t,k)}[1] \in \mathcal{R}^4, \tag{6}$$

$$\boldsymbol{B} = [\boldsymbol{u}_1|\boldsymbol{u}_2|...|\boldsymbol{u}_m|\boldsymbol{v}_1|\boldsymbol{v}_2|, ..., |\boldsymbol{v}_m]^T \in \mathcal{R}^{2M\times4}, \tag{7}$$

$$\boldsymbol{L}^{\text{3D}}_{(t,k)} = \Big(\boldsymbol{B}[:,:3]^T\ \boldsymbol{B}[:,:3]\Big)^{-1}\ \boldsymbol{B}[:,:3]^T\ (-\boldsymbol{B}[:,3]). \tag{8}$$

$\mathbb{M}_m[i,:]$ indicates the $i$-th row of $\mathbb{M}_m$. $\boldsymbol{B}[:,:i]$ indicates columns 0 to $i$-1 of $\boldsymbol{B}$. $\boldsymbol{B}[:,i]$ indicates the $i$-th column of $\boldsymbol{B}$. $\boldsymbol{L}_{(m,t,k)}[i]$ denotes the $i$-th element of $\boldsymbol{L}_{(m,t,k)}$.

Eq. (5) and Eq. (6) first compute the projection constraints for $\boldsymbol{L}^{\text{3D}}_{(t,k)}$, i.e., $\boldsymbol{u}_m[:3] \cdot \boldsymbol{L}^{\text{3D}}_{(t,k)} + \boldsymbol{u}_m[3] = 0$, where "$\cdot$" denotes the dot product and $\boldsymbol{u}_m[:3]$ denotes the first three elements of $\boldsymbol{u}_m$. Then we stack all the constraints into $\boldsymbol{B} \in \mathcal{R}^{2M\times4}$ in Eq. (7). Lastly, we solve for $\boldsymbol{L}^{\text{3D}}_{(t,k)}$ with a least squares approach in Eq. (8).

**3D Projection**. The second step for SBT is 3D projection. Given $\boldsymbol{L}^{\text{3D}}_{(t,k)}$, We calculate the reprojected 2D landmark in view$_m$, $\hat{\boldsymbol{L}}_{(m,t,k)}$, as follows:

$$\boldsymbol{q} = \mathbb{M}_m[:,:3]\boldsymbol{L}^{\text{3D}}_{(t,k)} + \mathbb{M}_m[:,3] \in \mathcal{R}^{3\times1}, \tag{9}$$

$$\hat{\boldsymbol{L}}_{(m,t,k)} = \begin{bmatrix} \boldsymbol{q}[0] \,/\, \boldsymbol{q}[2] \\ \boldsymbol{q}[1] \,/\, \boldsymbol{q}[2] \end{bmatrix} \in \mathcal{R}^{2\times1}, \tag{10}$$

**Loss calculation**. The SBT supervision comes from the discrepancy between the predicted coordinates $\boldsymbol{L}_{(m,t,k)}$ and the estimated coordinates $\hat{\boldsymbol{L}}_{(m,t,k)}$ from 3D triangulation and reprojection. The loss function that calculates this discrepancy can vary for regression and heatmap-based detectors. We will introduce our choices in Section 3.3.

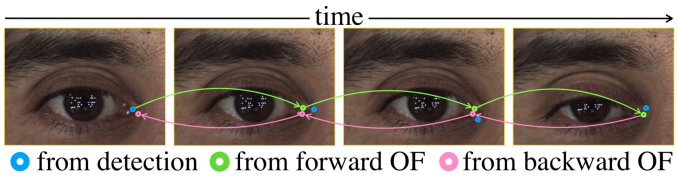



Fig. 4. **Forward-backward communication scheme** between the detector and the OF module during the training procedure. The green and pink lines indicate the forward and backward OF tracking routes. The blue/green/pink dots indicate the landmark predictions from the detector/forward-OF/backward-OF.

### 3.3 Supervision by Registration and Triangulation

SRT leverages supervision from three sources: manually labeled landmarks, registration, and triangulation. Figure 2 illustrates the procedure of SRT for human pose estimation. We now detail the loss function for each source of supervision.

#### 3.3.1 Supervision for Labeled Landmarks

Most detectors can be categorized into two different types. (I) Regression-based model takes an image $\boldsymbol{I}$ as input and directly regresses the coordinates of the facial landmarks $\boldsymbol{L}$ [9], [11]. (II) Heatmap-based model predicts for each landmark a heatmap $\boldsymbol{M}$, which encodes the confidence of the landmark being found at each location [12], [13]. For the regression-based model, an L1 loss was used to measure the error between the prediction $\boldsymbol{L}_{(m,t)}$ and ground truth $\boldsymbol{L}^*_{(m,t)}$:

$$\ell_{\text{det-L1}} = \sum\nolimits_{k=1}^{K} |\boldsymbol{L}_{(m,t,k)} - \boldsymbol{L}^*_{(m,t,k)}|_1. \tag{11}$$

For the heatmap-based model, we use the Hourglass model [13], and an L2 loss is applied to the predicted heatmap $\boldsymbol{M}$ and the ground truth heatmap $\boldsymbol{M}^*$ for each stage of the Hourglass model. The loss for a single stage is as follows:

$$\ell_{\text{det-L2}} = \sum\nolimits_{k=1}^{K} ||\boldsymbol{M}_{(m,t,k)} - \boldsymbol{M}^*_{(m,t,k)}||_F. \tag{12}$$

#### 3.3.2 Supervision-by-Registration

There are two main points to consider for the loss of SBR: (I) a suitable function to compare the discrepancy between two detections, and (II) removing incorrect supervision due to failures in optical flow or landmark detections. We propose a forward-backward communication scheme between the detection output and the OF module to tackle these two points. The forward communication computes the discrepancy while the backward communication evaluates the reliability of the OF module.

**Discrepancy comparison.** The registration loss directly computes the distance between the OF module's predictions ($\widetilde{\boldsymbol{L}}_{(m,t,k)}$, green dots in Figure 4) and the detector's predictions ($\boldsymbol{L}_{(m,t,k)}$, blue dots in Figure 4). The SBR loss for the regression-based model is as follows:

$$\ell_{\text{sbr-L1}} = \sum\nolimits_{k=1}^{K} \widetilde{\beta}_{(m,t,k)} ||\boldsymbol{L}_{(m,t,k)} - \widetilde{\boldsymbol{L}}_{(m,t,k)}||_1. \tag{13}$$

$\widetilde{\beta}_{(m,t,k)} \in \{0,1\}$ indicates the reliability of the $k$-th tracked landmark at time$_t$ in view$_m$, which is computed by the backward communication scheme detailed in the next section.

However, for heatmap-based detectors, we found that soft-$\arg\max$ [8] to extract $(x, y)$ location on heatmaps combined with $\ell_{\text{sbr-L1}}$ does not perform well in our experiments (see Table 6). Therefore, we propose a heatmap-based loss for the heatmap-based detector as follows:

$$\ell_{\text{sbr-L2}} = \sum\nolimits_{k=1}^{K} \widetilde{\beta}_{(m,t,k)} ||\boldsymbol{M}_{(m,t,k)} - \widetilde{\boldsymbol{M}}_{(m,t,k)}||_F, \tag{14}$$

where $\boldsymbol{M}_{(m,t,k)}$ indicates the heatmap of the $k$-th landmark at frame$_{(m,t)}$. $\widetilde{\boldsymbol{M}}_{(m,t,k)}$ is computed by warping the heatmap from the previous frame ($\boldsymbol{M}_{(m,t-1,k)}$) according to dense optical flow. $\ell_{\text{sbr-L2}}$ can effectively improve the accuracy of heatmap-based detectors.

**Removing incorrect supervision.** Inaccurate optical flow or landmark detections will lead to inaccurate supervision. The backward communication scheme, inspired by [43], filters out unreliable OF-generated landmarks from the SBR objective. It consists of three parts: (I) a landmark tracked from frame$_{(m,t-1)}$ to frame$_{(m,t)}$ and then from frame$_{(m,t)}$ to frame$_{(m,t-1)}$ should be consistent with its original coordinate. Otherwise, it means the OF module failed for this landmark. More specifically, $\widetilde{\beta}_{(m,t,k)} = 0$ if $||\boldsymbol{L}_{(m,t-1,k)} - G_{OF}(\boldsymbol{F}_{(m,t)}, \boldsymbol{F}_{(m,t-1)}, \widetilde{\boldsymbol{L}}_{(m,t,k)})|| > T_{\text{FB}}$, where $T_{\text{FB}}$ is the threshold for the forward-backward check. (II) the OF-generated results should not be far away from the detection results, i.e., $\widetilde{\beta}_{(m,t,k)} = 0$ if $||\widetilde{\boldsymbol{L}}_{(m,t,k)} - \boldsymbol{L}_{(m,t,k)}|| > T_{\text{D}}$, where $T_{\text{D}}$ is the threshold. (III) $\widetilde{\beta}_{(m,t,k)} = 0$ if the coordinates of detection or tracking results is outside the bounding box or image boundary. If the above three conditions do not hold, we deem the supervision to be reliable and set $\widetilde{\beta}_{(m,t,k)} = 1$.

#### 3.3.3 Supervision-by-Triangulation

The SBT loss is calculated as the distance between the detected landmark: $\boldsymbol{L}_{(m,t,k)}$, and the reprojected landmark: $\hat{\boldsymbol{L}}_{(m,t,k)}$, as follows:

$$\ell_{\text{sbt-L1}} = \sum\nolimits_{k=1}^{K} \hat{\beta}_{(m,t,k)} ||\boldsymbol{L}_{(m,t,k)} - \hat{\boldsymbol{L}}_{(m,t,k)}||_1. \tag{15}$$

For the heatmap-based model, we can use the soft-$\arg\max$ function to obtain $\boldsymbol{L}$ from $\boldsymbol{M}$ as in [8] and then apply Eq. (15). However, in experiments, this strategy often leads to unstable optimization (see Table 6). We thus consider the following objective for the heatmap-based model:

$$\ell_{\text{sbt-L2}} = \sum\nolimits_{k=1}^{K} \hat{\beta}_{(m,t,k)} ||\boldsymbol{M}_{(m,t,k)} - \hat{\boldsymbol{M}}_{(m,t,k)}||_F, \tag{16}$$

where $\hat{\boldsymbol{M}}_{(m,t,k)}$ is translated from $\boldsymbol{M}_{(m,t,k)}$ by the displacement $\hat{\boldsymbol{L}}_{(m,t,k)} - \boldsymbol{L}_{(m,t,k)}$.

Similar to SBR, it is also crucial that outliers are removed before applying SBT. Some projected landmarks result in a very large SBT loss, which suggests that these projected landmarks are likely to be in a wrong location. Using these landmarks in SBT could lead to incorrect supervision. To remove such outliers, we set $\hat{\beta}_{(m,t,k)} = 0$ if $||\hat{\boldsymbol{L}}_{(m,t,k)} - \boldsymbol{L}_{(m,t,k)}|| > T_{\text{TRI}}$, where $T_{\text{TRI}}$ is the threshold.

#### 3.3.4 Final Loss Function

The final loss is a weighted sum of all three kinds of supervisions as follows.

$$\ell = \ell_{\text{det}} + \omega_{\text{sbr}}\ell_{\text{sbr}} + \omega_{\text{sbt}}\ell_{\text{sbt}}, \tag{17}$$

TABLE 2
The description of 11 datasets used in our experiments. "HP" indicates human pose. "PF" and "PP" indicate Panoptic-Face and Panoptic-Pose.

| | modality | data type | #videos | #views | #images /frames | #landm -arks |
|---|---|---|---|---|---|---|
| 300-W [10], [44], [45] | image | face | N/A | 1 | 3837 | 68 |
| AFLW [7] | image | face | N/A | 1 | 24386 | 19 |
| WFLW [22] | image | face | N/A | 1 | 10000 | 98 |
| 300-VW [46], [47], [48] | video | face | 114 | 1 | 218597 | 68 |
| PF [49], [50] | video | face | 81 | 7 | 172415 | 70 |
| VoxCeleb2 [51] | video | face | 29970 | 1 | 2997000 | N/A |
| MPII [52] | image | HP | N/A | 1 | 40522 | 16 |
| PP [49], [50] | video | HP | 275 | 9 | 825000 | 19 |
| Human-3.6M [53] | video | HP | 836 | 4 | 2103096 | 32 |
| Mugsy-V1 | video | face | 441 | 6 | 196222 | 18 |
| Synthetic-Face | image | face | N/A | 1 | 3543 | 18 |

where $\ell_{\text{det}}$, $\ell_{\text{sbr}}$, and $\ell_{\text{sbt}}$ represent the detection, SBR, and SBT losses. $\omega_{\text{sbr}}$ and $\omega_{\text{sbt}}$ are weights of the corresponding losses.

## 4 Experimental Analysis

### 4.1 Datasets

To demonstrate the effectiveness of SRT, we performed experiments with 11 datasets summarized in Table 2.

#### 4.1.1 Single-View Image Landmark Detection Datasets

**300-W** [10], [44], [45] is a facial landmark detection dataset, which contains more than 3000 facial images with 68 landmarks. We use the common setting on 300-W [8], [9], [17], in which the dataset is split into the training set, the common test set, the challenge test set, and the full test set.

**AFLW** [7] is another facial landmark detection dataset, which contains 24386 facial images with 19 landmarks. We follow [8], [9] and split the whole dataset into the training set, the frontal test set, and the full test set. We also create two different test sets: one only containing images with yaw degree lower than $30^o$ and another only containing images with yaw degree larger than $60^o$.

**WFLW** [22] contains 7500 training images and 2500 test images with 98 facial landmarks.

**MPII** [52] is a human pose estimation dataset, containing 28821 training images and 11701 test images. This dataset has annotations for 16 body joints. We use the official training, validation, and test splits.

**Synthetic-Face** [8] is an internal synthetic face dataset. It contains 1770 training images and 1773 test images from the same person. This dataset is synthetically generated from a 3D model of a person's face. As a result, we can guarantee the annotations of this dataset have zero error, thus by perturbing these accurate annotations, we can measure the effect of inaccuracies in annotations. Sample images are shown in the supplementary material.

#### 4.1.2 Single-View Video Landmark Detection Datasets

**300-VW** [46], [47], [48] contains 50 training videos with 95192 frames. It provides three test sets: A has 31 videos with 62135 frames, B has 19 videos with 32805 frames, C has 14 videos with 26338 frames.

**VoxCeleb2** [51] contains thousands of videos of people talking. It does not have landmark annotation, and we thus use it as an unlabeled video dataset for enhancing detectors with SBR. Since it is very large, we only sample 5 videos for each identity and extract the first 100 frames of each video.

#### 4.1.3 Multi-view Landmark Detection Datasets

**CMU-Panoptic** [49] is collected from a massively synchronized and geometrically calibrated multi-view system, which contains more than 30 high-definition views (1920×1080 resolution). In this dataset, they provide 3D facial landmark labels for some videos. By only keeping frames where the frontal face is visible and choosing the first 2000 frames of each video, we construct a new dataset named Panoptic-Face (PF). Similar to Panoptic-Face, we use the multi-view videos with labeled human pose information as a new dataset for pose estimation, named Panoptic-Pose.

**Human-3.6M** [53] is collected in a space with the size of 3 meters × 4 meters. 4 views from 4 different synchronized cameras are used. We use the subjects 1, 5, 6, 7, 8, 9, 11 as the unlabeled multi-view videos to enhance the detector.

**Mugsy-V1** is an internal multi-view facial video dataset. It contains 6 views and 441 videos in total. There are 196222 frames in total, and 12543 frames are annotated with 18 facial landmarks. We use 6394 annotated frames as the test set, and the remaining 6149 annotated frames together with 183679 unlabeled frames as the training set. Note that (I) all annotated frames are in the same view, and (II) the training frames are not from the same video as any test frames. When we validate the hyper-parameters, we randomly select 50% training samples for the validation set.

### 4.2 Experiment Settings

**Data Augmentation.** We use the same data augmentation for most experiments unless otherwise specified. Given a bounding box, we sequentially apply six steps. (I) Expand this box by 20% height and width. (II) Randomly resize this box in the range of [90%, 110%] with probability of 50%. (III) Randomly translate this box, in which the maximum displacement of the box center is 10% height or width. (IV) Randomly rotate this box by the maximum 40 degrees. (V) Crop a face/body image by this box and convert this image into gray-scale. (VI) Apply intensity augmentation on this cropped image by multiplying one random scalar in [0.6, 1.4] to each channel.

**The Setup of Regression-based Model.** Due to the high efficiency of MobileNet-V2 [54], we choose it as the backbone of our regression-based detector. We made some modification based on MobileNet-V2 and its final layer is an fully connected layer, which outputs a vector with the dimension of $2 \times K$. In all experiments, we train this regression-based detector via Adam [55]. We train the detector for 200 epochs with the initial learning rate as 0.001. We reduce the learning rate by a factor of 10 at 100-th and 150 epochs. For facial datasets, we resize the input face into 96×96 by default, but use 240×320 for Mugsy-V1. We found that using the restart technique [56] can significantly improve the performance of the regression-based model. Specifically, we train the model for 200 epochs, and then we increase the learning rate to 0.1 and retrain the model for another 200 epochs. We repeat

TABLE 3
We show the effect of utilizing various unlabeled data sets to enhance the regression-based detector. We report NME and $P\text{-}error$ on three test subsets of 300-W. We also report NME, AUC@0.08, and $P\text{-}error$ on 300-VW A, B, and C. $\omega_{\text{sbr}}$ and $\omega_{\text{sbt}}$ indicate the loss weights of SBR and SBT supervision, respectively. $\times$ denotes the corresponding supervision is not used. For all experiments in this table, we use the 49 landmarks, excluding landmarks on the facial boundary. The SBR/SBT/SRT numbers are averaged over 3 runs.

| Labeled Data | Unlabeled Data | $\omega_{\text{sbr}}$ | $\omega_{\text{sbt}}$ | 300-W Test Set | | | 300-VW A | | | 300-VW B | | | 300-VW C | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | Common | Challenge | Full | NME | AUC | $P\text{-}error$ | NME | AUC | $P\text{-}error$ | NME | AUC | $P\text{-}error$ |
| 300-W Training Set | N/A | × | × | 2.99 / 12.08‰ | 5.83 / 21.44‰ | 3.55 / 13.97‰ | 4.58 | 53.76 | 15.27‰ | 3.98 | 52.81 | 15.46‰ | 9.31 | 45.90 | 24.03‰ |
| | 300-VW | 0.1 | × | 2.86 / 10.73‰ | 5.56 / 19.82‰ | 3.39 / 12.39‰ | 4.39 | 55.00 | 13.76‰ | 3.76 | 54.48 | 13.84‰ | 9.19 | 48.63 | 21.90‰ |
| | 300-VW | 0.5 | × | 2.86 / 9.99‰ | 5.55 / 19.88‰ | 3.39 / 11.47‰ | 4.12 | 57.43 | 12.59‰ | 3.61 | 56.00 | 12.51‰ | 8.47 | 52.57 | 19.97‰ |
| | 300-VW | 1.0 | × | 2.87 / 9.74‰ | 5.50 / 19.79‰ | 3.39 / 11.15‰ | 4.10 | 57.94 | 12.36‰ | 3.62 | 56.22 | 12.12‰ | 8.53 | 53.12 | 19.51‰ |
| | 300-VW | 2.0 | × | 2.93 / 9.67‰ | 5.59 / 18.54‰ | 3.45 / 11.06‰ | 4.16 | 57.45 | 12.31‰ | 3.67 | 55.41 | 11.75‰ | 8.39 | 53.45 | 19.10‰ |
| | VoxCeleb2 | 0.1 | × | 2.87 / 10.49‰ | 5.46 / 18.25‰ | 3.37 / 11.78‰ | 4.39 | 53.97 | 13.29‰ | 3.88 | 52.80 | 13.56‰ | 8.64 | 48.55 | 20.66‰ |
| | VoxCeleb2 | 0.5 | × | 2.85 / 10.04‰ | 5.43 / 16.92‰ | 3.36 / 11.16‰ | 4.32 | 54.80 | 12.94‰ | 3.85 | 53.07 | 12.90‰ | 8.61 | 47.15 | 18.77‰ |
| | VoxCeleb2 | 1.0 | × | 2.86 / 9.91‰ | 5.58 / 17.29‰ | 3.39 / 11.05‰ | 4.34 | 53.86 | 13.33‰ | 3.87 | 52.79 | 13.16‰ | 8.61 | 43.17 | 18.88‰ |
| | VoxCeleb2 | 2.0 | × | 2.90 / 10.43‰ | 5.69 / 18.13‰ | 3.44 / 11.65‰ | 4.59 | 51.80 | 15.26‰ | 3.94 | 51.83 | 14.38‰ | 9.28 | 35.81 | 20.88‰ |
| | PF | 0.1 | × | 2.87 / 10.73‰ | 5.49 / 20.24‰ | 3.38 / 12.23‰ | 4.25 | 55.22 | 13.27‰ | 3.76 | 53.96 | 13.54‰ | 8.70 | 50.09 | 21.12‰ |
| | PF | 0.5 | × | 2.87 / 10.58‰ | 5.59 / 19.76‰ | 3.40 / 12.17‰ | 4.27 | 55.33 | 13.78‰ | 3.83 | 53.46 | 14.32‰ | 8.50 | 50.18 | 22.76‰ |
| | PF | 1.0 | × | 2.88 / 10.43‰ | 5.58 / 21.06‰ | 3.41 / 12.05‰ | 4.35 | 54.71 | 14.47‰ | 3.92 | 52.33 | 15.28‰ | 8.90 | 45.75 | 24.88‰ |
| | PF | 2.0 | × | 2.95 / 10.81‰ | 5.89 / 25.90‰ | 3.53 / 12.90‰ | 5.60 | 50.76 | 18.20‰ | 4.48 | 49.19 | 19.35‰ | 15.82 | 24.54 | 37.69‰ |
| | PF | × | 0.1 | 2.86 / 10.84‰ | 5.39 / 19.31‰ | 3.35 / 12.12‰ | 4.19 | 54.60 | 13.24‰ | 3.79 | 53.86 | 13.58‰ | 8.59 | 48.02 | 21.23‰ |
| | PF | × | 0.5 | 2.86 / 10.63‰ | 5.39 / 18.23‰ | 3.35 / 11.93‰ | 4.08 | 55.72 | 12.09‰ | 3.76 | 53.85 | 12.79‰ | 8.85 | 48.56 | 19.10‰ |
| | PF | × | 1.0 | 2.88 / 10.31‰ | 5.47 / 17.98‰ | 3.38 / 11.68‰ | 4.11 | 55.21 | 11.60‰ | 3.87 | 52.56 | 12.51‰ | 8.77 | 46.35 | 18.60‰ |
| | PF | × | 2.0 | 2.94 / 10.70‰ | 5.53 / 18.17‰ | 3.45 / 12.11‰ | 4.25 | 54.01 | 11.65‰ | 3.99 | 51.19 | 12.78‰ | 9.22 | 40.27 | 18.49‰ |
| | PF | 0.1 | 0.1 | 2.85 / 10.56‰ | 5.42 / 19.59‰ | 3.36 / 11.93‰ | 4.15 | 55.41 | 12.85‰ | 3.77 | 54.20 | 13.42‰ | 8.21 | 49.91 | 20.18‰ |
| | PF | 0.5 | 0.5 | 2.87 / 10.13‰ | 5.49 / 18.03‰ | 3.39 / 11.72‰ | 4.09 | 55.90 | 11.88‰ | 3.83 | 53.04 | 12.83‰ | 8.44 | 46.60 | 19.12‰ |
| | PF | 0.5 | 1.0 | 2.90 / 10.45‰ | 5.52 / 19.12‰ | 3.41 / 11.75‰ | 4.17 | 54.70 | 11.79‰ | 3.95 | 51.72 | 13.06‰ | 8.78 | 42.61 | 18.71‰ |
| | PF | 1.0 | 1.0 | 2.92 / 10.36‰ | 5.72 / 19.15‰ | 3.47 / 11.99‰ | 4.36 | 52.80 | 12.00‰ | 4.09 | 50.31 | 13.45‰ | 9.31 | 34.13 | 19.78‰ |
| | PF | 2.0 | 2.0 | 3.01 / 11.20‰ | 6.56 / 23.45‰ | 3.71 / 13.16‰ | 6.68 | 45.89 | 18.48‰ | 5.49 | 46.55 | 17.54‰ | 18.55 | 18.75 | 30.24‰ |

TABLE 4
Results of SBR and SBT to enhance the regression-based detector on Mugsy-V1. $\times$ denotes the corresponding supervision is not used. Note that since Mugsy-V1 includes an unlabeled portion in the training set, no other unlabeled datasets are used to train the detectors.

| $\omega_{\text{sbr}}$ | $\omega_{\text{sbt}}$ | Validation | | | Test | | |
|---|---|---|---|---|---|---|---|
| | | NME | AUC | $P\text{-}error$ | NME | AUC | $P\text{-}error$ |
| × | × | 4.18 | 48.79 | 4.96‰ | 4.63 | 43.47 | 5.14‰ |
| 0.1 | × | 4.11 | 49.62 | 5.07‰ | 4.53 | 44.45 | 5.06‰ |
| 0.5 | × | 4.09 | 50.01 | 4.91‰ | 4.48 | 45.10 | 4.86‰ |
| 1.0 | × | 4.05 | 50.37 | 4.77‰ | 4.44 | 45.48 | 4.71‰ |
| 2.0 | × | 4.07 | 50.43 | 4.78‰ | 4.47 | 45.17 | 4.82‰ |
| × | 0.1 | 4.06 | 49.67 | 4.51‰ | 4.48 | 44.94 | 4.60‰ |
| × | 0.5 | 4.02 | 50.21 | 3.82‰ | 4.39 | 45.92 | 3.80‰ |
| × | 1.0 | 4.01 | 50.25 | 3.42‰ | 4.42 | 45.48 | 3.38‰ |
| × | 2.0 | 4.14 | 48.71 | 3.09‰ | 4.60 | 43.39 | 3.05‰ |
| 0.5 | 0.5 | 3.85 | 52.02 | 3.45‰ | 4.36 | 46.30 | 3.68‰ |
| 1.0 | 0.5 | 3.85 | 52.00 | 3.47‰ | 4.38 | 46.14 | 3.70‰ |
| 2.0 | 0.5 | 3.87 | 51.87 | 3.40‰ | 4.38 | 46.14 | 3.69‰ |
| 0.5 | 1.0 | 3.90 | 51.51 | 3.17‰ | 4.43 | 45.49 | 3.39‰ |
| 1.0 | 1.0 | 3.92 | 51.21 | 3.25‰ | 4.42 | 45.61 | 3.33‰ |
| 2.0 | 1.0 | 3.91 | 51.38 | 3.09‰ | 4.44 | 45.39 | 3.33‰ |

this procedure twice on 300-W and AFLW, and repeat it ten times on Mugsy-V1.

**The Setup of Heatmap-based Model.** We use the stacked hourglass model [13] as our heatmap-based detector. We use four hourglass stages and set the recursive step as 3. The number of channels of the intermediate features is 256. We follow the official training strategy introduced in [13]. We use the RMSProp optimizer and an initial learning rate of 0.001. For face datasets, we train the model for 160 epochs and decay the learning rate by a factor of 2 at the 60-th, 90-th, 110-th, and 130-th epoch. We resize the input face into 256×256 for the heatmap-based detector.

**Training Procedure.** We assume the bounding box for face or human is already obtained. During training, the input is always a cropped face for facial landmark detection or human body for pose estimation. We utilized a stage wise training strategy with two steps: (1) optimizing the detector with detection loss only; and (2) enhancing the detector with detection, SBR, and SRT losses jointly.

**Training with SBR.** We perform the OF tracking over three consecutive frames by default. For differential LK OF, the $\Omega$ in Eq. (1) is a $13 \times 13$ patch centered at the landmark. The maximum iterations of OF is 20 and the convergence threshold for $\Delta \mathbf{p} = 10^{-6}$. For the input feature of the OF module, we use the gray-scale image. In the forward-backward communication scheme, $T_{\text{FB}}$ and $T_{\text{D}}$ are both 1% of the square root of the area of the face/body bounding box. When using the bilinear interpolation OF, we use the Farneback algorithm [57] to compute dense optical flow.

**Training with SBT.** We perform 3D triangulation on four views, i.e., $M = 4$. Specifically, given an image in one view, we first randomly sample three different views from the current view. We use the corresponding landmark predictions in these four views to obtain the 3D coordinates, and then project 3D coordinates to 2D coordinates in these four views. The threshold $T_{\text{TRI}}$ is 1% of the square root of the area of the face/body bounding box.

**Balanced sampling of labeled and unlabeled data per batch.** The amount of unlabeled data is oftentimes much larger than the amount of labeled data. Therefore, we explicitly ensure that the network is seeing a balanced amount of labeled and unlabeled samples, such that the network does not "forget" human supervision. Specifically, in one batch, we have 32 labeled images, 32 video triplets (each contains three consecutive frames), and 16 multi-view quadruplets (each contains images from four different views).

**Evaluating accuracy**. We measured the Normalized Mean Error (NME) and the Area Under the Curve (AUC)@0.08

TABLE 5
We utilize various unlabeled data sets to enhance the regression-based detector on different AFLW test sets. yaw$\leq 30^o$: all test faces with the yaw degree lower than 30 degrees. yaw$\geq 60^o$: all test faces with the yaw degree higher than 60 degrees. Our approach significantly improves the precision (reducing $P\text{-}error$) compared to the baseline models. The SBR/SBT/SRT numbers are averaged over 3 runs.

| Labeled Data | Unlabeled Data | $\omega_{\text{sbr}}$ | $\omega_{\text{sbt}}$ | AFLW-Front | | | AFLW-Full | | | yaw$\leq 30^o$ | | | yaw$\geq 60^o$ | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | NME | AUC | $P\text{-}error$ | NME | AUC | $P\text{-}error$ | NME | AUC | $P\text{-}error$ | NME | AUC | $P\text{-}error$ |
| AFLW Training Set | N/A | × | × | 1.89 | 77.25 | 19.49‰ | 2.65 | 68.07 | 12.98‰ | 2.12 | 74.07 | 14.82‰ | 3.94 | 53.94 | 31.30‰ |
| | 300-VW | 0.1 | × | 1.80 | 77.78 | 18.10‰ | 2.58 | 68.69 | 11.93‰ | 2.07 | 74.58 | 13.53‰ | 3.83 | 54.64 | 30.05‰ |
| | 300-VW | 0.5 | × | 1.81 | 77.64 | 17.65‰ | 2.58 | 68.60 | 11.35‰ | 2.08 | 74.48 | 13.10‰ | 3.86 | 54.49 | 29.93‰ |
| | 300-VW | 1.0 | × | 1.82 | 77.54 | 17.52‰ | 2.60 | 68.49 | 11.21‰ | 2.08 | 74.39 | 12.90‰ | 3.88 | 54.31 | 29.65‰ |
| | 300-VW | 2.0 | × | 1.85 | 77.21 | 18.06‰ | 2.62 | 68.18 | 12.14‰ | 2.11 | 74.06 | 13.59‰ | 3.89 | 53.95 | 29.18‰ |
| | VoxCeleb2 | 0.1 | × | 1.81 | 77.74 | 17.85‰ | 2.58 | 68.64 | 11.78‰ | 2.07 | 74.53 | 13.45‰ | 3.82 | 54.58 | 29.42‰ |
| | VoxCeleb2 | 0.5 | × | 1.81 | 77.65 | 17.88‰ | 2.58 | 68.59 | 12.02‰ | 2.07 | 74.49 | 13.60‰ | 3.81 | 54.44 | 29.14‰ |
| | VoxCeleb2 | 1.0 | × | 1.83 | 77.49 | 18.87‰ | 2.60 | 68.42 | 14.63‰ | 2.10 | 74.35 | 15.23‰ | 3.84 | 54.14 | 28.65‰ |
| | VoxCeleb2 | 2.0 | × | 1.86 | 77.09 | 19.63‰ | 2.64 | 67.96 | 15.60‰ | 2.14 | 73.91 | 15.87‰ | 3.88 | 53.60 | 28.97‰ |
| | PF | 0.1 | × | 1.81 | 77.70 | 18.13‰ | 2.57 | 68.62 | 11.97‰ | 2.07 | 74.49 | 13.67‰ | 3.81 | 54.49 | 29.74‰ |
| | PF | 0.5 | × | 1.82 | 77.59 | 17.79‰ | 2.59 | 68.48 | 11.53‰ | 2.09 | 74.36 | 13.32‰ | 3.85 | 54.39 | 29.35‰ |
| | PF | 1.0 | × | 1.83 | 77.49 | 17.81‰ | 2.61 | 68.26 | 11.71‰ | 2.10 | 74.24 | 13.45‰ | 3.85 | 54.12 | 29.40‰ |
| | PF | 2.0 | × | 1.88 | 76.97 | 18.65‰ | 2.67 | 67.63 | 12.32‰ | 2.14 | 73.72 | 14.16‰ | 3.96 | 53.21 | 30.89‰ |
| | PF | × | 0.1 | 1.82 | 77.69 | 18.16‰ | 2.58 | 68.67 | 11.97‰ | 2.08 | 74.55 | 13.76‰ | 3.84 | 54.59 | 29.90‰ |
| | PF | × | 0.5 | 1.83 | 77.54 | 17.88‰ | 2.59 | 68.51 | 11.50‰ | 2.09 | 74.36 | 13.31‰ | 3.83 | 54.48 | 29.19‰ |
| | PF | × | 1.0 | 1.84 | 77.39 | 17.90‰ | 2.62 | 68.21 | 11.54‰ | 2.11 | 74.11 | 13.45‰ | 3.90 | 54.13 | 29.38‰ |
| | PF | × | 2.0 | 1.87 | 77.05 | 18.76‰ | 2.67 | 67.68 | 12.28‰ | 2.15 | 73.74 | 14.14‰ | 3.97 | 53.17 | 30.71‰ |
| | PF | 0.1 | 0.1 | 1.81 | 77.68 | 17.92‰ | 2.58 | 68.66 | 11.70‰ | 2.07 | 74.48 | 13.50‰ | 3.83 | 54.65 | 29.72‰ |
| | PF | 0.5 | 0.5 | 1.84 | 77.37 | 17.78‰ | 2.61 | 68.29 | 11.59‰ | 2.11 | 74.17 | 13.51‰ | 3.85 | 54.27 | 28.95‰ |
| | PF | 0.5 | 1.0 | 1.84 | 77.31 | 18.17‰ | 2.62 | 68.10 | 11.82‰ | 2.11 | 74.02 | 13.78‰ | 3.90 | 53.86 | 29.93‰ |
| | PF | 1.0 | 1.0 | 1.85 | 77.19 | 18.34‰ | 2.65 | 67.91 | 12.15‰ | 2.12 | 73.95 | 13.88‰ | 3.95 | 53.62 | 30.06‰ |
| | PF | 2.0 | 2.0 | 1.90 | 76.64 | 19.21‰ | 2.71 | 67.18 | 12.73‰ | 2.17 | 73.33 | 14.69‰ | 4.03 | 52.62 | 30.85‰ |

metric. We use the inter-ocular distance on 300-W and the face size on AFLW for normalization. When reporting NME and AUC, we omit the % notation. For pose estimation, we use the standard Percentage of Correct Keypoints by a fraction of the head size (PCKh) for evaluation.

**Evaluating precision**. We propose to utilize the Equivariant Landmark Transformation (ELT) [58], which was originally used as a loss function, as a proxy to measure the precision of the detector. ELT applies a known affine transformation to an image, and we would expect the landmark detections on both the original image and the transformed image to just differ by the affine transformation. If the detections do not completely follow the affine transformation, this is a sufficient condition for the detector not being very precise, i.e., the detections are not consistent across images. Therefore, we use this as a proxy to measure the precision of the detector. More specifically, given an image $\boldsymbol{I}$, we apply two separate random data augmentation transformations[2] and obtain $\boldsymbol{I}^a$ and $\boldsymbol{I}^b$. This procedure can be formulated as $\boldsymbol{I}^a = \text{Affine}_{\Theta^a}(\boldsymbol{I})$ and $\boldsymbol{I}^b = \text{Affine}_{\Theta^b}(\boldsymbol{I})$, where $\text{Affine}_{\Theta^a}$ denotes the affine transformation parametrized by $\Theta^a$. By giving $\boldsymbol{I}^a$ and $\boldsymbol{I}^b$ into the detector, we obtain landmark predictions $\boldsymbol{L}_p^a$ and $\boldsymbol{L}_p^b$ respectively for landmark $p$. We then compute the following metric to evaluate precision:

$$P\text{-}error = \frac{1}{\eta}\sum\nolimits_k ||\text{Affine}_{\Theta^a}^{-1}(\boldsymbol{L}_k^a) - \text{Affine}_{\Theta^b}^{-1}(\boldsymbol{L}_k^b)||, \quad (18)$$

where $P\text{-}error$ (Precision-error) is calculated as the mean discrepancy of each point pair $\boldsymbol{L}_p^a$ and $\boldsymbol{L}_p^b$ once the inverse affine transform has been applied to the points. $\eta$ is a normalization constant, which is the square root of the area of the bounding box. The advantage of ELT is that no additional labeled data is necessary, thus $P\text{-}error$ is not affected by the inconsistencies in human annotations.

2. We apply three steps: (I) randomly scale the bounding box in a range of [0.8, 1.2], (II) randomly translate the box with the maximum displacement being 10% height horizontally and 10% width vertically, (III) randomly rotate the box with the maximum degree of 30.

TABLE 6
The effect of different loss functions for detectors on Mugsy-V1.

| Losses | Validation | | | Test | | |
|---|---|---|---|---|---|---|
| | NME | AUC | $P\text{-}error$ | NME | AUC | $P\text{-}error$ |
| the regression-based detector | | | | | | |
| – | 4.18 | 48.79 | 4.96‰ | 4.63 | 43.47 | 5.14‰ |
| L1 on $\boldsymbol{L}$ for SBR | 4.05 | 50.37 | 4.77‰ | 4.44 | 45.48 | 4.71‰ |
| L2 on $\boldsymbol{L}$ for SBR | 4.13 | 49.40 | 5.02‰ | 4.53 | 44.65 | 5.14‰ |
| L1 on $\boldsymbol{L}$ for SBT | 4.01 | 50.25 | 3.42‰ | 4.42 | 45.48 | 3.38‰ |
| L2 on $\boldsymbol{L}$ for SBT | 4.08 | 49.58 | 4.75‰ | 4.49 | 44.90 | 4.70‰ |
| L1 on $\boldsymbol{L}$ for SRT | 3.85 | 52.02 | 3.45‰ | 4.36 | 46.30 | 3.68‰ |
| the heatmap-based detector | | | | | | |
| – | 3.78 | 52.78 | 2.65‰ | 4.04 | 50.05 | 2.60‰ |
| L1 on $\boldsymbol{L}$ for SBR | 6.37 | 21.83 | 3.29‰ | 7.67 | 12.12 | 7.12‰ |
| L2 on $\boldsymbol{L}$ for SBR | 3.95 | 51.79 | 2.93‰ | 3.97 | 50.88 | 2.33‰ |
| L2 on $\boldsymbol{M}$ for SBR | 3.77 | 52.85 | 2.35‰ | 3.96 | 51.02 | 2.31‰ |
| L1 on $\boldsymbol{L}$ for SBT | 5.36 | 33.60 | 0.79‰ | 5.54 | 31.77 | 0.73‰ |
| L2 on $\boldsymbol{L}$ for SBT | 3.78 | 52.85 | 2.29‰ | 3.98 | 50.87 | 2.20‰ |
| L2 on $\boldsymbol{M}$ for SBT | 3.77 | 52.98 | 2.24‰ | 3.80 | 50.88 | 2.20‰ |
| L2 on $\boldsymbol{M}$ for SRT | 3.76 | 52.98 | 2.10‰ | 3.94 | 51.20 | 2.08‰ |

### 4.3 Ablation Studies

We did a series of experiments to study the effect of (I) different kinds of optical flow algorithms for approximation. (II) different weights for $\omega_{\text{sbr}}$ and $\omega_{\text{sbt}}$; (III) different kinds of supervision; (IV) different kinds of data source; (V) different kinds of loss; (VI) annotation noise; and (VII) effectiveness of removing incorrect supervision. Since the regression-based detector is more efficient and simple than the heatmap-based detector, we use the regression-based detector for most ablation studies.

TABLE 7
The comparison of the normalized mean error (NME) on AFLW. "HB" indicates the heatmap-based detector.

| Methods | CCL [15] | SAN [17] | Wing [59] | HB | HB + SRT |
|---|---|---|---|---|---|
| AFLW-Full | 2.72 | 1.91 | 1.47 | 2.31 | 2.26 |
| AFLW-Front | 2.17 | 1.85 | - | 1.71 | 1.64 |

TABLE 8
PCKh with a threshold of 0.5 evaluation metric on the MPII Human Pose validation set.

| Methods | SRT | Params (M) | PCKh | $P$-$error$ |
|---|---|---|---|---|
| CU-Net [60] | – | 24.18 | 71.51 | 119.21‰ |
| CPM [12] | – | 25.40 | 72.12 | 92.27‰ |
| MSPN [61] | – | 26.06 | 76.30 | 89.39‰ |
| heatmap-based [13] | – | 25.26 | **80.50** | 84.42‰ |
| heatmap-based | Panoptic-Pose | 25.26 | 80.03 | 83.17‰ |
| heatmap-based | Human-3.6M | 25.26 | 80.15 | **83.05‰** |

TABLE 9
The comparison of NME w.r.t. the inter-ocular distance on 300-W. We also compute $P$-$error$ on the full test set of 300-W.

| Methods | Common | Challenging | Full Set | $P$-$error$ |
|---|---|---|---|---|
| 68 landmarks | | | | |
| Two-Stage [9] | 4.36 | 7.42 | 4.96 | - |
| Pose-Invariant [62] | 5.43 | 9.88 | 6.30 | - |
| PCD-CNN [63] | 3.67 | 7.62 | 4.44 | - |
| SBR [8] | 3.28 | 7.58 | 4.10 | - |
| SAN [17] | 3.34 | 6.60 | 3.98 | 17.41 |
| LAB [22] | 2.98 | 5.19 | 3.49 | - |
| ODN [64] | 3.56 | 6.67 | 4.17 | - |
| heatmap-based | 2.81 | 5.50 | 3.34 | 16.15 |
| heatmap-based + ELT | 2.81 | 5.53 | 3.33 | 16.01 |
| heatmap-based + SRT | 2.80 | 5.61 | 3.39 | 15.23 |
| 49 landmarks (without landmarks on the facial contour) | | | | |
| SAN [17] | 2.42 | 5.42 | 3.01 | 13.10 |
| regression-based | 2.99 | 5.83 | 3.55 | 13.97 |
| regression-based + SRT | 2.84 | 5.36 | 3.31 | 10.89 |
| heatmap-based | 2.00 | 3.93 | 2.38 | 9.94 |
| heatmap-based + SRT | 1.98 | 3.99 | 2.41 | 8.87 |

**The effect of different approximation methods for OF.** As discussed in Section 3.1.2, we could approximate OF at sub-pixel locations through bilinear interpolation on a pre-computed flow field. We empirically compare the difference between the actual OF results and the interpolation results. On the first 200 frames of all 14 videos in the 300-VW C test set, we use the ground truth coordinate of each landmark from the previous frame as initial location, and compute the tracked results at the current frame. There is only 0.02 average pixel discrepancy between the interpolated results and the actual OF results. This shows that the OF computed from bilinear interpolation is very accurate. We further evaluated seven different OF methods (please see supplementary material for details). Different OF methods yield similar performance, but the differentiable Lucas Kanade OF [8] is significantly slower than utilizing bilinear approximation (e.g., more than 41 times slower than Farneback [57]). Since the differentiable Lucas Kanade OF takes too much time, we leverage the interpolation strategy for our remaining experiments. We choose Farneback [57] as our default OF algorithm because (1) it is one of the most popular methods, (2) it achieves similar performance than others, (3) its speed is acceptable.

**The effect of different loss weights: $\omega_{\text{sbr}}$, $\omega_{\text{sbt}}$.** We tested different loss weights on Mugsy-V1 in Table 4. Among $\{0.1, 0.5, 1.0, 2.0\}$ for $\omega_{\text{sbr}}$, $\omega_{\text{sbr}}$ of 1.0 gives the highest AUC and lowest NME on the validation set. Different $\omega_{\text{sbr}}$ gives similar $P$-$error$. Among $\{0.1, 0.5, 1.0, 2.0\}$ for $\omega_{\text{sbt}}$, a higher $\omega_{\text{sbt}}$ will result in a lower $P$-$error$, which means the model is precise, i.e., more robust to spatial transformation. However, the training AUC drops when $\omega_{\text{sbt}}$ becomes too large, i.e., the detectors becomes more precise but also more inaccurate. This could be due to the triangulation constraint forcing the detector to predict a landmark at a consistent-in-3D but wrong location. Therefore, in order to balance both NME, AUC, and $P$-$error$, we use $\omega_{\text{sbr}} = 1.0$ for SBR, and $\omega_{\text{sbt}} = 1.0$ for SBT. For SRT (SBR+SBT), we use $\omega_{\text{sbr}} = 0.5$, $\omega_{\text{sbt}} = 0.5$.

**Comparing SBR and SBT.** Table 3 shows SBR and SBT results on 300-W. Several conclusions can be made: (I) temporal information from 300-VW, VoxCeleb2, and Panoptic-Face can improve the detector's accuracy. (II) multi-view information from Panoptic-Face can improve the detector's accuracy. (III) the mutual benefit from both temporal and multi-view cues can further enhance the detection performance. (IV) using the multi-view cue of Panoptic-Face obtains lower NME on the image test set than using the temporal cue. Table 5 and Table 4 shows the results on AFLW and Mugsy-V1 respectively. We can observe similar phenomena as Table 3. Table 4 also shows that multi-view information is more beneficial to precision improvement than temporal information.

**The effect of different kinds of the unlabeled data source.** We try different unlabeled datasets in Table 3 and Table 5. All four different unlabeled datasets can enhance the base detector, while there are some interesting observations. (I) Unlabeled 300-VW provides the best NME and AUC results on 300-VW A test category compared to others. (II) Even if VoxCeleb2 is 10 times larger than 300-VW, "300-W + VoxCeleb2" obtains a worse result than "300-W + 300-VW". The faces in VoxCeleb2 are blurred and are different from 300-VW. In contrast, the faces in the 300-VW training set are somewhat similar to that of the 300-VW test set w.r.t. pose, motion, etc. (III) Unlabeled Panoptic-Face does not improves the detector as much as 300-VW or VoxCeleb2. This might be due to the limited diversity (number of identities) of Panoptic-Face.

**The effect of different kinds of the losses** is analyzed in Table 6. We test different kinds loss functions when enhancing a detector with SBR and SBT. When using the regression-based detector, both "L1 on $\boldsymbol{L}$" and "L2 on $\boldsymbol{L}$" (L1, L2 loss on coordinates) can significantly improve the base detector. L1-based SBR and SBT obtain a better performance compared to L2-based SBR and SBT, thus we selected "L1 on $\boldsymbol{L}$" for regression-based detector for SBR and SBT.

When using the heatmap-based detector, there are three different loss functions: "L1 on $\boldsymbol{L}$", "L2 on $\boldsymbol{L}$", and "L2 on $\boldsymbol{M}$" (heatmaps). SBR and SBT with "L1 on $\boldsymbol{L}$" do not work well[3]. "L2 on $\boldsymbol{M}$" gives a slightly lower NME and higher

3. In Table 6, "L1 on $\boldsymbol{L}$" for SBT may have very good $P$-$error$, but NME and AUC are very poor. This is an example of a detector being precise but inaccurate.

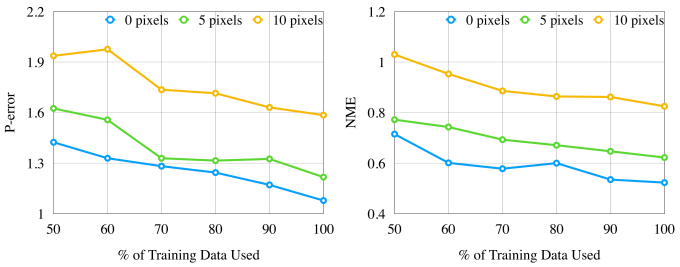



Fig. 5. The effect of data size and noise level for the heatmap-based model on the test set of Synthetic-Face. We randomly add Gaussian noise with std=0,5,10 pixels to the ground truth labels of the training set.

TABLE 10
Results on three 300-VW test sets. Two different sources of bounding box detections are used as input. HB + SRT means heatmap-based detector trained with SRT. Most results come from [65], [66].

| Methods | | 300-VW A | | 300-VW B | | 300-VW C | |
|---|---|---|---|---|---|---|---|
| Detection | Landmark | AUC | Failure | AUC | Failure | AUC | Failure |
| DPM | CFSS [67] | 76.6 | 3.74 | 77.0 | 1.32 | 72.4 | 5.23 |
| | ERT [68] | 77.7 | 3.44 | 77.2 | 1.51 | 72.1 | 6.08 |
| | HB + SRT | 78.1 | 2.32 | 77.8 | 1.31 | 74.0 | 4.98 |
| MTCNN | CFSS [67] | 73.4 | 8.51 | 72.5 | 8.52 | 72.6 | 5.69 |
| | STCSR [69] | 79.1 | 2.40 | 78.8 | 0.32 | 71.0 | 4.46 |
| | HB + SRT | 74.3 | 7.37 | 73.7 | 7.16 | 73.2 | 4.91 |

AUC than "L2 on $\boldsymbol{L}$". Therefore, we choose the loss "L2 on $\boldsymbol{M}$" for the heatmap-based detector.

**The effect of inaccurate annotations.** We perturb the training data of Synthetic-Face with various amounts of Gaussian noise, and train the heatmap-based detector 3 times with different random seeds on varying amounts of training data (50% to 100%) and noise levels (standard deviation 0, 5, and 10). Results shown in Figure 5 suggests the following: (I) Training with more data can improve both precision and accuracy. (II) Training with more accurate labels can improve both precision and accuracy. For example, having 100% of the training data with std=10 pixel noise leads to worse performance than having just 50% of the training data but no error in labels. (III) NME and $P\text{-}error$ are highly correlated. The advantage of $P\text{-}error$ is that it does not require annotations thus will not be affected by inaccurate annotations. This makes it a good proxy for performance when testing set accuracy has saturated.

**Effectiveness of removing incorrect supervision.** Here, we quantitatively show how much supervision are regarded as incorrect and removed. The percentage of $\widetilde{\beta}_{(m,t,k)}$ and $\hat{\beta}_{(m,t,k)}$ that are set to 0, i.e. supervision removed, varies based on the datasets. We show an example of SRT enhancing the regression-based detector on AFLW. When using 300-VW with SBR, the zero percentage of $\widetilde{\beta}_{(m,t,k)}$ is about 3.16%. When using Panoptic-Face with SBR, the zero percentage of $\widetilde{\beta}_{(m,t,k)}$ is about 8.95%. When using Panoptic-Face with SBT only, the zero percentage is about 32.91%. When using Panoptic-Face with SRT, zero percentage of $\widetilde{\beta}_{(m,t,k)}$ and $\hat{\beta}_{(m,t,k)}$ are about 9.15% and 51.91%, respectively.

However, the removal of incorrect supervision is not always perfect. One common failure case is false positives in the forward-backward consistency check. For a landmark, if the detector makes the same mistake on two consecutive frames, and if the optical flow between the two detections are also consistent, then this will pass the check and be incorrectly used as supervision in SRT. To quantitatively measure the frequency of these failure cases, we run SRT with our 300-W base detector on 300-VW and leverage the available per-frame labels of facial-landmarks to measure the failure rate. Landmarks that pass the forward-backward consistency check but have more than 0.05 normalized distance w.r.t. the ground truth labels are considered as false positives. Results show that 0.46% landmarks suffer from false positives in the consistency check, i.e., our method can filter out more than 99% of the incorrect supervision.

TABLE 11
Comparison results on WFLW [22]. "PF" indicates Panoptic-Face.

| Methods | | | NME | AUC@0.1 | Failure@0.1 | $P\text{-}error$ |
|---|---|---|---|---|---|---|
| LAB [22] | | | 5.27 | 53.23 | 7.56 | 17.21 |
| base model | SBR | SBT | NME | AUC@0.1 | Failure@0.1 | $P\text{-}error$ |
| regression-based | – | – | 6.72 | 41.88 | 14.16 | 22.16 |
| regression-based | 300-VW | – | 6.55 | 43.00 | 13.76 | 13.97 |
| regression-based | PF | – | 6.64 | 41.94 | 14.24 | 20.35 |
| regression-based | PF | PF | 6.54 | 42.61 | 13.48 | 20.27 |
| heatmap-based | – | – | 5.33 | 53.04 | 8.44 | 18.05 |
| heatmap-based | 300-VW | – | 5.13 | 54.59 | 7.28 | 16.36 |
| heatmap-based | PF | – | 5.13 | 54.65 | 7.08 | 15.49 |
| heatmap-based | PF | PF | 5.13 | 54.64 | 7.07 | 15.01 |

### 4.4 Comparison with the State-of-the-art

We compare our algorithm with other state-of-the-art algorithms on both facial and body landmark detection. For facial landmark detection, the unlabeled data used for SRT are from the 300-VW training set, VoxCeleb2 and Panoptic-Face. For body landmark detection, the unlabeled data used for SRT are from Panoptic-Pose and Human-3.6M.

**Results on 300-W** are shown in Table 9. We ran SRT on 68 landmarks of 300-W, which harms the heatmap detector. This is because some landmarks, such as facial contour, are ambiguous across different views. By filtering out such outlier, we observe a significant improvement based on the regression-based detector. There is still no clear improvement when SRT is applied to the heatmap-based detector.

**Results on 300-VW**. We compare with some baseline methods from [65] in Table 10. For each face bounding box detection algorithm used, our method is competitive with state-of-the-art methods.

**Results on AFLW** are shown in Table 7. SRT improves the heatmap-based detector by about 4% w.r.t. NME for AFLW-Front.

**Results on WFLW**. We show that SRT can significantly improve the precision of both regression-based and heatmap-based detectors in Table 11. They also slightly reduce the NME and increase the AUC. With the assistance of SRT, we achieve competitive results compared to LAB [22].

**Results on MPII**. We show results of our SRT on the MPII pose estimation dataset in Table 8, which reports PCKh@0.5 of the single crop evaluation setting on the validation set. For a fair comparison, we tune some structure hyper-parameters of different models to make the number of parameters similar. Unfortunately, the performance drops compared to the base detector. There could be two reasons:

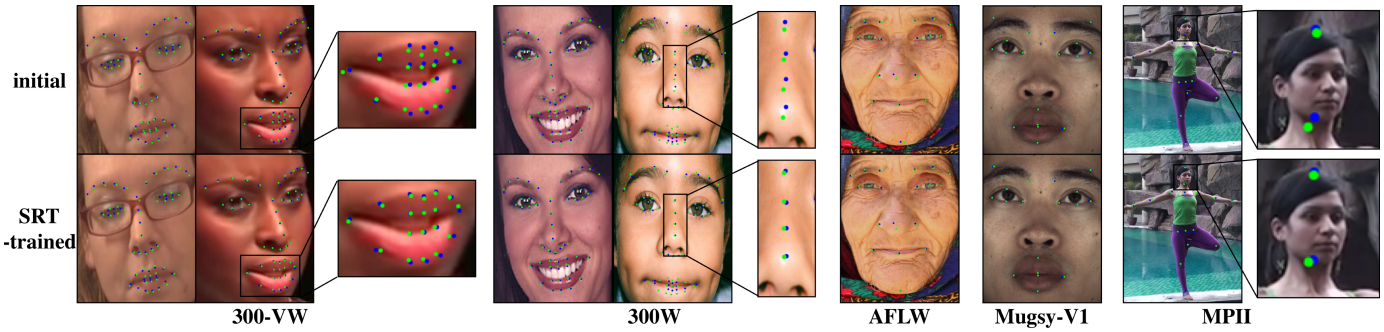



Fig. 6. Visualization results of the regression-based models on 300-W, 300-VW, AFLW, and Mugsy-V1, and the heatmap-based models on MPII. Green and blue points are prediction results and ground truth labels, respectively.

(I) hyper-parameters should be carefully tuned on different datasets, such as SBR/SBT weights and forward-backward communication thresholds; and (II) the diversity of unlabeled multi-view videos is not enough to obtain a good performance, since there are only nine unlabeled multi-view videos in Panoptic-Pose.

**Discussion**. For heatmap-based detectors, SRT improved performance on Mugsy-V1 (Table 6), WFLW (Table 11), and AFLW (Table 7), while performance did not improve on MPII (Table 8) and 300-W (Table 9). We hypothesize that the incorrect supervision that was not removed by our checks (see Section 4.3) could have led to the drop in performance. The heatmap-based detector has powerful representation ability, thus it could remember both (I) the good supervision from the labeled data and (II) the bad supervision from the failure cases of registration and triangulation which our filtering method could not filter out. Thus the generalization ability is reduced by those failure cases leading to degraded performance on the test set. On the other hand, SRT still improves the heatmap detectors performance on AFLW, WFLW, and Mugsy-V1. One reason could be because the number of training images in AFLW and WFLW is much larger than that of 300-W, thus leading to a better initial landmark location prediction as input to SRT. In sum, SRT may not always lead to accuracy improvements, but we still see consistent improvement in precision.

## 5 Conclusion and Future Work

In this manuscript, we propose SRT, an unsupervised method to improve a regression or heatmap-based landmark detection model by leveraging registration and triangulation supervision, which does not require any additional manual annotations. Our conclusions are as follows. (I): both registration and triangulation can improve accuracy and precision, and they complement each other. (II): there is no clear winner to which supervision, registration or triangulation, is more effective. (III): bilinear optical flow interpolation for registration works as well as differentiable Lucas Kanade optical flow [8] while providing much faster training speed. (IV): the unlabeled data used should have a similar distribution as the labeled training and testing data, and large amounts of out-of-domain data is not as effective as smaller amounts of in-domain data. (V): false positives of the forward-backward check is a common way that SRT might degrade the detector's performance. This is due to the SRT losses striving for consistency, i.e. precision, which might not necessarily coincide with the correctness of the detection, i.e. accuracy. (VI): $P\text{-}error$ could be a good proxy for performance especially when testing set accuracy has saturated, as $P\text{-}error$ is not affected by inconsistencies in annotations. However, a low $P\text{-}error$ does not necessarily mean good accuracy, so $P\text{-}error$ should not be used in a vacuum.

In order to further push this research forward, one key ingredient missing is a large-scale, high resolution, multi-camera dataset of faces and bodies with high quality annotations for a large number of subjects under different environments. The current multi-view landmark detection datasets have a large number of video frames but a small number of subjects. Such limited subjects are insufficient for both robust training and convincing evaluation. Also, investigating how to achieve high annotation quality is also crucial in both training and evaluation of detectors. In addition to the dataset, we suggest designing new evaluation protocols for landmark detection to test a detector's robustness w.r.t. the quality of images, geometric calibration, and so on. While the accuracy of landmark detection has been boosted again and again, little effort has been devoted to the study of robustness.

## References

[1] C. Cao, Y. Weng, S. Lin, and K. Zhou, "3D shape regression for real-time facial animation," *ACM Transactions on Graphics*, vol. 32, no. 4, p. 41, 2013.

[2] C. Wu, T. Shiratori, and Y. Sheikh, "Deep incremental learning for efficient high-fidelity face tracking," in *ACM SIGGRAPH Asia*, vol. 37, no. 6, 2018, pp. 234:1–234:12.

[3] J. S. Yoon, T. Shiratori, S.-I. Yu, and H. S. Park, "Self-supervised adaptation of high-fidelity face models for monocular performance tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4601–4609.

[4] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Nießner, "Face2Face: Real-time face capture and reenactment of RGB videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2387–2395.

[5] H. Joo, T. Simon, and Y. Sheikh, "Total capture: A 3d deformation model for tracking faces, hands, and bodies," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8320–8329.

[6] R. Alp Güler, N. Neverova, and I. Kokkinos, "Densepose: Dense human pose estimation in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7297–7306.

[7] M. Koestinger, 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 *Proc. IEEE Int. Conf. Comput. Vis. Workshop*, 2011.

[8] X. Dong, S.-I. Yu, X. Weng, S.-E. Wei, Y. Yang, and Y. Sheikh, "Supervision-by-Registration: An unsupervised approach to improve the precision of facial landmark detectors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 360–368.

[9] J. Lv, X. Shao, J. Xing, C. Cheng, and X. Zhou, "A deep regression architecture with two-stage reinitialization for high performance facial landmark detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3317–3326.

[10] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: The first facial landmark localization challenge," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop*, 2013.

[11] X. Xiong and F. De la Torre, "Supervised descent method and its applications to face alignment," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 532–539.

[12] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 4724–4732.

[13] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 483–499.

[14] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, "Rmpe: Regional multi-person pose estimation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2334–2343.

[15] S. Zhu, C. Li, C.-C. Loy, and X. Tang, "Unconstrained face alignment via cascaded compositional learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3409–3417.

[16] P. Dollár, P. Welinder, and P. Perona, "Cascaded pose regression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 1078–1085.

[17] X. Dong, Y. Yan, W. Ouyang, and Y. Yang, "Style aggregated network for facial landmark detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 379–388.

[18] X. Yu, F. Zhou, and M. Chandraker, "Deep deformation network for object landmark localization," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 52–70.

[19] Y. Li, B. Sun, T. Wu, and Y. Wang, "Face detection with end-to-end integration of a convnet and a 3D model," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 420–436.

[20] A. Bulat and G. Tzimiropoulos, "How far are we from solving the 2D & 3D face alignment problem? (and a dataset of 230,000 3d facial landmarks)," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1021–1030.

[21] X. Nie, J. Feng, J. Xing, S. Xiao, and S. Yan, "Hierarchical contextual refinement networks for human pose estimation," *IEEE Trans. Image Process.*, vol. 28, no. 2, pp. 924–936, 2019.

[22] W. Wu, C. Qian, S. Yang, Q. Wang, Y. Cai, and Q. Zhou, "Look at boundary: A boundary-aware face alignment algorithm," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2129–2138.

[23] S. Baker and I. Matthews, "Lucas-Kanade 20 years on: A unifying framework," *Int. J. Comput. Vis.*, vol. 56, no. 3, pp. 221–255, 2004.

[24] M. H. Khan, J. McDonagh, and G. Tzimiropoulos, "Synergy between face alignment and tracking via discriminative global consensus optimization," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017.

[25] H. Liu, J. Lu, J. Feng, and J. Zhou, "Two-stream transformer networks for video-based face alignment," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 11, pp. 2546–2554, 2018.

[26] X. Peng, R. S. Feris, X. Wang, and D. N. Metaxas, "A recurrent encoder-decoder network for sequential face alignment," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 38–56.

[27] R. Girdhar, G. Gkioxari, L. Torresani, M. Paluri, and D. Tran, "Detect-and-track: Efficient pose estimation in videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 350–359.

[28] X. Peng, S. Zhang, Y. Yu, and D. N. Metaxas, "Toward personalized modeling: Incremental and ensemble alignment for sequential faces in the wild," *Int. J. Comput. Vis.*, vol. 126, no. 2-4, pp. 184–197, 2018.

[29] X. Zhu, "Semi-supervised learning tutorial," in *Proc. Int. Conf. Machine Learning*, 2007, pp. 1–135.

[30] J. Charles, T. Pfister, D. Magee, D. Hogg, and A. Zisserman, "Personalizing human video pose estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3063–3072.

[31] H. Shen, S.-I. Yu, Y. Yang, D. Meng, and A. Hauptmann, "Unsupervised video adaptation for parsing human motion," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 347–360.

[32] T. Simon, H. Joo, I. Matthews, and Y. Sheikh, "Hand keypoint detection in single images using multiview bootstrapping," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1145–1153.

[33] S. Suwajanakorn, N. Snavely, J. J. Tompson, and M. Norouzi, "Discovery of latent 3D keypoints via end-to-end geometric reasoning," in *Proc. Advances Neural Inf. Process. Syst.*, 2018, pp. 2063–2074.

[34] Y. Jafarian, Y. Yao, and H. S. Park, "Monet: Multiview semi-supervised keypoint via epipolar divergence," *arXiv preprint arXiv:1806.00104*, 2018.

[35] Y. Zhang and H. S. Park, "Multiview supervision by registration," *arXiv preprint arXiv:1811.11251*, 2018.

[36] H. Rhodin, J. Spörri, I. Katircioglu, V. Constantin, F. Meyer, E. Müller, M. Salzmann, and P. Fua, "Learning monocular 3d human pose estimation from multi-view images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8437–8446.

[37] B. Amberg, A. Blake, A. Fitzgibbon, S. Romdhani, and T. Vetter, "Reconstructing high quality face-surfaces using model based stereo," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2007, pp. 1–8.

[38] D. Pavllo, C. Feichtenhofer, D. Grangier, and M. Auli, "3d human pose estimation in video with temporal convolutions and semi-supervised training," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7753–7762.

[39] B. D. Lucas, T. Kanade *et al.*, "An iterative image registration technique with an application to stereo vision," in *Proc. Int. Joint Conf. on AI*, 1981, pp. 674–679.

[40] C.-H. Chang, C.-N. Chou, and E. Y. Chang, "CLKN: Cascaded lucas-kanade networks for image alignment," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2213–2221.

[41] M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, "Spatial transformer networks," in *Proc. Advances Neural Inf. Process. Syst.*, 2015, pp. 2017–2025.

[42] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge University Press, 2003.

[43] Z. Kalal, K. Mikolajczyk, and J. Matas, "Forward-backward error: Automatic detection of tracking failures," in *Proc. IEEE Int. Conf. Pattern Recognit.*, 2010, pp. 2756–2759.

[44] C. Sagonas, E. Antonakos, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: Database and results," *Image and Vision Computing*, vol. 47, pp. 3–18, 2016.

[45] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "A semi-automatic methodology for facial landmark annotation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshop*, 2013.

[46] G. G. Chrysos, E. Antonakos, S. Zafeiriou, and P. Snape, "Offline deformable face tracking in arbitrary videos," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop*, 2015.

[47] J. Shen, S. Zafeiriou, G. G. Chrysos, J. Kossaifi, G. Tzimiropoulos, and M. Pantic, "The first facial landmark tracking in-the-wild challenge: Benchmark and results," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop*, 2015.

[48] G. Tzimiropoulos, "Project-out cascaded regression with an application to face alignment," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3659–3667.

[49] H. Joo, T. Simon, X. Li, H. Liu, L. Tan, L. Gui, S. Banerjee, T. Godisart, B. Nabbe, I. Matthews *et al.*, "Panoptic studio: A massively multiview system for social interaction capture," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 1, pp. 190–204, 2019.

[50] H. Joo, H. Liu, L. Tan, L. Gui, B. Nabbe, I. Matthews, T. Kanade, S. Nobuhara, and Y. Sheikh, "Panoptic studio: A massively multiview system for social motion capture," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3334–3342.

[51] J. S. Chung, A. Nagrani, and A. Zisserman, "VoxCeleb2: Deep speaker recognition," in *Proc. of the Conference of the International Speech Communication Association*, 2018, pp. 1086–1090.

[52] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2D human pose estimation: New benchmark and state of the art analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 3686–3693.

[53] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 7, pp. 1325–1339, 2014.

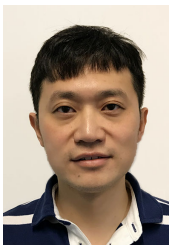
[54] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4510–4520.

[55] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015.

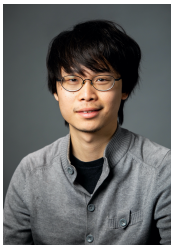
[56] I. Loshchilov and F. Hutter, "Sgdr: Stochastic gradient descent with warm restarts," in *Proc. Int. Conf. Learn. Representations*, 2017.

[57] G. Farnebäck, "Two-frame motion estimation based on polynomial expansion," in *Scandinavian Conference on Image Analysis*, 2003, pp. 363–370.
[58] S. Honari, P. Molchanov, S. Tyree, P. Vincent, C. Pal, and J. Kautz, "Improving landmark localization with semi-supervised learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1546–1555.
[59] Z.-H. Feng, J. Kittler, M. Awais, P. Huber, and X.-J. Wu, "Wing loss for robust facial landmark localisation with convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2235–2245.
[60] Z. Tang, X. Peng, S. Geng, Y. Zhu, and D. Metaxas, "Cu-net: Coupled u-nets," in *Proc. of the British Machine Vision Conference*, 2018.
[61] W. Li, Z. Wang, B. Yin, Q. Peng, Y. Du, T. Xiao, G. Yu, H. Lu, Y. Wei, and J. Sun, "Rethinking on multi-stage networks for human pose estimation," *arXiv preprint arXiv:1901.00148*, 2019.
[62] A. Jourabloo, M. Ye, X. Liu, and L. Ren, "Pose-invariant face alignment with a single cnn," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3219–3228.
[63] A. Kumar and R. Chellappa, "Disentangling 3d pose in a dendritic cnn for unconstrained 2d face alignment," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 430–439.
[64] M. Zhu, D. Shi, M. Zheng, and M. Sadiq, "Robust facial landmark detection via occlusion-adaptive deep networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3486–3496.
[65] G. G. Chrysos, E. Antonakos, P. Snape, A. Asthana, and S. Zafeiriou, "A comprehensive performance evaluation of deformable face tracking in-the-wild," *Int. J. Comput. Vis.*, vol. 126, no. 2-4, pp. 198–232, 2018.
[66] J. Deng, G. Trigeorgis, Y. Zhou, and S. Zafeiriou, "Joint multi-view face alignment in the wild," *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3636–3648, 2019.
[67] S. Zhu, C. Li, C. Change Loy, and X. Tang, "Face alignment by coarse-to-fine shape searching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 4998–5006.
[68] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 1867–1874.
[69] J. Yang, J. Deng, K. Zhang, and Q. Liu, "Facial shape tracking via spatio-temporal cascade shape regression," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop*, 2015.

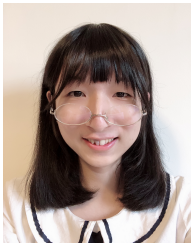
**Xuanyi Dong** received the B.E. degree in Computer Science and Technology from Beihang University, Beijing, China, in 2016. He is currently a Ph.D. student at at School of Computer Science, University of Technology Sydney, Australia. His research interests include automated deep learning and its application to real world applications.



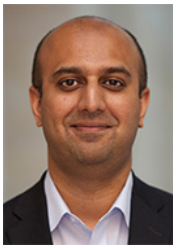
**Yi Yang** received the Ph.D. degree in computer science from Zhejiang University, Hangzhou, China, in 2010. He was a post-doctoral researcher with the School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA. He is currently a Professor with University of Technology Sydney, Australia. His current research interest includes machine learning and its applications to multimedia content analysis and computer vision.



**Shih-En Wei** is currently a research scientist focused on machine learning and computer vision for social VR in Facebook Reality Labs, Pittsburgh, Pennsylvania. Prior to that, he received the M.S. in Robotics from the School of Computer Science, Carnegie Mellon University, Pittsburgh. He also received his B.S. in Electrical Engineering and M.S. degree in Communication Engineering from National Taiwan University, Taipei, Taiwan. His research interests mainly lie in tracking human's behavior including eyes, face, hands, and body poses.



**Xinshuo Weng** received the BSc degree from Wuhan University, China, and the MS degree from Carnegie Mellon University. She is currently a Ph.D. student at the Robotics Institute of Carnegie Mellon University. Before starting her Ph.D. program, she was working at Oculus Research Pittsburgh (now Facebook Reality Lab) as a research engineer. Her research interests include computer vision and machine learning, with a special interest in 3D vision and self-supervised learning.



**Yaser Sheikh** directs the Facebook Reality Lab in Pittsburgh, which is focused on achieving metric telepresence in AR and VR, and is an adjunct professor at Carnegie Mellon University. His research broadly focuses on machine perception and rendering of social behavior, spanning sub-disciplines in computer vision, computer graphics, and machine learning. With colleagues and students, he has won the Honda Initiation Award (2010), Popular Sciences "Best of Whats New" Award, best student paper award at CVPR (2018), best paper finalist awards at (CVPR 2019), best paper awards at WACV (2012), SAP (2012), SCA (2010), ICCV THEMIS (2009), best demo award at ECCV (2016), and he received the Hillman Fellowship for Excellence in Computer Science Research (2004). Yaser has served as a senior committee member at leading conferences in computer vision, computer graphics, and robotics including SIGGRAPH (2013, 2014), CVPR (2014, 2015, 2018), ICRA (2014, 2016), ICCP (2011), and serves as an Associate Editor of TPAMI. His research has been featured by various media outlets including The New York Times, BBC, MSNBC, Popular Science, and in technology media such as WIRED, The Verge, and New Scientist.

**Shoou-I Yu** is currently a research scientist focused on machine learning and computer vision for social VR in Facebook Reality Labs, Pittsburgh, Pennsylvania. Prior to that, he received the Ph.D. in Language Technologies from the School of Computer Science, Carnegie Mellon University, Pittsburgh. He also received the B.S. in Computer Science and Information Engineering from National Taiwan University, Taipei, Taiwan. His research interests mainly lie in landmark detection, multi-object tracking, and multimedia retrieval.