Real-Time Facial Segmentation and Performance Capture from RGB Input

Shunsuke Saito Tianye Li Hao Li

Pinscreen University of Southern California

Abstract. We introduce the concept of unconstrained real-time 3D facial performance capture through explicit semantic segmentation in the RGB input. To ensure robustness, cutting edge supervised learning approaches rely on large training datasets of face images captured in the wild. While impressive tracking quality has been demonstrated for faces that are largely visible, any occlusion due to hair, accessories, or hand-to-face gestures would result in significant visual artifacts and loss of tracking accuracy. The modeling of occlusions has been mostly avoided due to its immense space of appearance variability. To address this curse of high dimensionality, we perform tracking in unconstrained images assuming non-face regions can be fully masked out. Along with recent breakthroughs in deep learning, we demonstrate that pixel-level facial segmentation is possible in real-time by repurposing convolutional neural networks designed originally for general semantic segmentation. We develop an efficient architecture based on a two-stream deconvolution network with complementary characteristics, and introduce carefully designed training samples and data augmentation strategies for improved segmentation accuracy and robustness. We adopt a state-of-the-art regression-based facial tracking framework with segmented face images as training, and demonstrate accurate and uninterrupted facial performance capture in the presence of extreme occlusion and even side views. Furthermore, the resulting segmentation can be directly used to composite partial 3D face models on the input images and enable seamless facial manipulation tasks, such as virtual make-up or face replacement.

Keywords: real-time facial performance capture, face segmentation, deep convolutional neural network, regression

1 Introduction

Recent advances in real-time 3D facial performance capture [1,2,3,4,5,6,7] have not only transformed the entertainment industry with highly scalable animation and affordable production tools [8], but also popularized mobile social media apps with facial manipulation. Many state-of-the-art techniques have been developed to operate robustly in natural environments, but pure RGB solutions are still susceptible to occlusions (e.g., caused by hair, hand-to-face gestures, or accessories), which result in unpleasant visual artifacts or the inability to correctly initialize facial tracking.

While it is known that the shape and appearance of fully visible faces can be represented compactly through linear models [9,10], any occlusion or uncontrolled illumination could cause high non-linearities to a 3D face fitting problem.

As this space of variation becomes intractable, supervised learning methods have been introduced to predict facial shapes through large training datasets of face images captured under unconstrained and noisy conditions. We observe that if such *occlusion noise* can be fully eliminated, the dimensionality of facial modeling could be drastically reduced to that of a well-posed and constrained problem. In other words, if reliable dense facial segmentation is possible, 3D facial tracking from RGB input becomes a significantly easier problem. Only recently has the deep learning community demonstrated highly effective semantic segmentations, such as the fully convolutional network (FCN) of [11] or the deconvolutional network (DeconvNet) of [12], by repurposing highly efficient classification networks [13,14] for dense predictions of general objects (e.g., humans, cars, etc.).

We present a real-time facial performance capture approach by explicitly segmenting facial regions and processing masked RGB data. We rely on the effectiveness of deep learning to achieve clean facial segmentations in order to enable robust facial tracking under severe occlusions. We propose an end-to-end segmentation network that also uses a two-stream deconvolution network with complementary characteristics, but shares the lower convolution network to enable real-time performance. A final convolutional layer recombines both outputs into a single probability map which is converted into a refined segmentation mask via graph cut algorithm [15]. Our 3D facial tracker is based on a state-of-the-art displaced dynamic expression (DDE) method [5] trained with segmented input data. Separating facial regions from occluding objects with similar colors and fine structures (e.g. hands) is extremely challenging, even for existing segmentation network. We propose a training data augmentation strategy based on perturbations, croppings, occlusion generation, hand compositings, as well as the use of negative samples containing no faces. Once our dense prediction model is trained, we replace the training database for DDE regression with masked faces obtained from our convolutional network.

We demonstrate uninterrupted tracking in the presence of highly challenging occlusions such as hands which have similar skin tones as the face and fine scale boundary details. Furthermore, our facial segmentation enables interesting compositing effects such as tracked facial models under hair and other occluding objects. These capabilities were only demonstrated recently using a robust geometric model fitting approach on depth sensor data [7].

We make the following contributions:

- We present the first real-time facial segmentation framework from pure RGB input using a convolutional neural network. We demonstrate the importance of carefully designed datasets and data augmentation strategies for handling challenging occlusions such as hands.
- We improve the efficiency and accuracy of existing segmentation networks using an architecture based on two-stream deconvolution networks and shared convolution network.
- We demonstrate superior tracking accuracy and robustness through explicitly facial segmentation and regression with masked training data, and outperform the current state-of-the-art.

2 Related Work

The fields of facial tracking and animation have undergone a long thread of major research milestones in both, the vision and graphics community, as well as influencing the industry widely over the past two decades.

In high-end film and game production, performance-driven techniques are commonly used to scale the production of realistic facial animation. An overview is discussed in Pighin and Lewis [16]. To meet the high quality bars, techniques for production typically build on sophisticated sensor equipments and controlled capture settings [17,18,19,20,21,22,23,24]. While exceptional tracking accuracy can be achieved, these methods are generally computationally expensive and the full visibility of the face needs to be ensured.

On the other extreme, 2D facial tracking methods that work in fully unconstrained settings have been explored extensively for applications such as face recognition and emotion analytics. Even though only sparse 2D facial landmarks are detected, many techniques are designed to be robust to uncontrolled poses, challenging lighting conditions, and rely on a single-view 2D input. Early algorithms are based on parametric models [25,26,27,28,29], but later outperformed by more robust and real-time data-driven methods such as active appearance models (AAM) [30] and constrained local models (CLM) [31]. While the landmark mean-shift approach of [32] and the supervised descent method of [33] avoid the need of user-specific training, more efficient solutions exist based on explicit shape regressions [34,35,36]. However, these methods are all sensitive to occlusions and only a limited number of 2D features can be detected.

Weise and colleagues [37] demonstrated the first system to produce compelling facial performance capture in real-time using a custom 3D depth sensor based on structured light. The intensive training procedure was later reduced significantly using an example-based algorithm developed by Li and collaborators [38]. With consumer depth sensors becoming mainstream (e.g., Kinect, Realsense, etc.), a whole line of real-time facial animation research have been developed with focus on deployability. The work of [1] incorporated pre-recorded motion priors to ensure stable tracking for noisy depth maps, which resulted in the popular animation software, Faceshift [8]. By optimizing the identity and expression models online, Li and coworkers [3], as well as Bouaziz and collaborators [2] eliminated the need of user-specific calibration. For uninterrupted tracking under severe occlusions, Hsieh and colleagues [7] recently proposed an explicit facial segmentation technique, but requires a depth sensor.

While the generation of 3D facial animations from pure RGB input have been demonstrated using sparse 2D landmarks detection [39,40,41], a superior performance capture fidelity and robustness has only been shown recently by Cao and coworkers [4] using a 3D shape regression approach. Cao and colleagues [5] later extended the efficient two-level boosted regression technique introduced in [34] to the 3D case in order to avoid user-specific calibration. Higher fidelity facial tracking from monocular video has also been demonstrated with additional high-resolution training data [6], very large datasets of a person [42], or more expensive non-real-time computation [43,44]. While robust to unconstrained lighting environments and large head poses, these methods are sensitive to large occlusions and cannot segment facial regions.

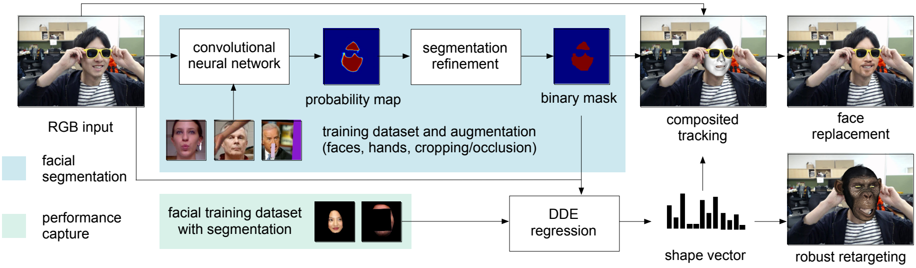
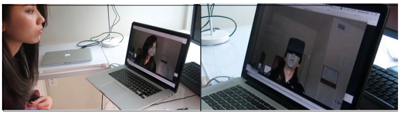


Fig. 1: Overview of our facial segmentation and performance capture pipeline.

Due to the immense variation of facial appearances in unconstrained images, it is extremely challenging to obtain clean facial segmentations at the pixel level. The hierarchical CNN-based parsing network of Luo and collaborators [45] generates masks of individual facial components such as eyes, nose, and mouth even in the presence of occlusions, but does not segment the facial region as a whole. Smith and coworkers [46] use an example-based approach for facial region and component segmentation, but the method requires sufficient visibility of the face. These two methods are computationally intensive and susceptible to wrong segmentations when occlusions have similar colors as the face. By alternating between face mask prediction and landmark localization with deformable part models, Ghiasi and Fowlkes [47] have recently demonstrated state-of-the-art facial segmentation results on the Caltech Occluded Faces in the Wild (COFW) dataset [48] at the cost of expensive computations. Without explicitly segmenting the face, occlusion handling methods have been proposed for the detection of 2D landmarks within an AAM frameworks [49], but superior results were later shown using techniques based on discriminatively trained deformable parts model [50,51]. Highly efficient landmark detection has been recently demonstrated using cascade of regressors trained with occlusion data [48,52].

3 Overview


capture setting using a laptop-integrated RGB camera

As illustrated in Figure 1, our system is divided into a facial segmentation stage (blue) and a performance capture stage (green). Our pipeline takes an RGB image as input and produces a binary segmentation mask in addition to a tracked 3D face model, which is parameterized by a shape vector, as output. The binary mask represents a per-pixel facial region estimated by a deep learning framework for facial segmentation. Following Cao et al.'s DDE regression technique [5], the shape vector describes the rigid head motion and the facial expression coefficients, which drive the animation of a personalized 3D tracking model. In addition, the shape of the user's identity and the focal length are solved concurrently during performance capture. While the resulting tracking model represents the shape of the subject, the shape vector can be used to retarget any digital character with compatible animation controls as input.

Our convolutional neural network first predicts a probability map on a cropped rectangular face region for which size and positions are determined based on the bounding box of the projected 3D tracking model from the previous frame. The face region of the initial frame is detected using the method of Viola and Jones [53]. The output probability map is a smaller fixed-size resolution image (128 × 128 pixels) and describes the likelihood for each pixel being labeled as part of the specific face region. While two output maps (one for the overall shape and one for fine-scaled details) are simultaneously produced by our two-stream deconvolution network, a single output probability map is generated through a final convolutional layer. To ensure accurate and robust facial segmentation, we train our convolutional neural network using a large dataset of segmented face images, augmented with peturbations, synthetic occlusions, croppings, and hand compositings, as well as negative samples containing no faces. We convert the resulting probability map into a binary mask using a graph cut algorithm [54] and bilinearly upsample the mask to the original input resolution.

We then use this segmentation mask as input to the facial tracker as well as for compositing partial 3D facial models during occlusions. This facial segmentation technique is also used to produce training data for the regression model of the DDE framework. Our facial performance capture pipeline is based on the state-of-the-art method of [5], which does not require any calibration step for individual users. The training process and the regression explicitly take the segmentation mask into account. Our system runs in real-time on commercially available desktop machines with sufficiently powerful GPU processors. For many mobile devices such as laptops, which are not yet ready for deep neural net computations, we can optionally offload the segmentation processing over Wi-Fi to a desktop machine with high-end GPU resources for real-time performance.

4 Facial Segmentation

Our facial segmentation pipeline computes a binary mask from the bounding box of a face in the input image. The cropped face image is first resized to a small 128 × 128 pixel resolution image, which is passed to a convolutional neural network for a dense 2-class segmentation problem. Similar to state-of-the-art segmentation networks [11,12,55], the overall network consists of two parts, (1) a lower convolution network for multi-dimensional feature extraction and (2) a higher deconvolution network for shape generation. This shape corresponds to the segmented object and is reconstructed using the features obtained from the convolution network. The output is a dense 128 × 128 probability map that assigns each pixel to either a face or non-face region. While both state-of-the-art networks, FCN [11] and DeconvNet [12] use the identical convolutional network based on VGG-16 layers [56], they approach deconvolution differently. FCN performs a simple deconvolution using a single bilinear interpolation layer, and produces coarse, but clean overall shape segmentations, because the output layer is closely connected to the convolution layers preventing the loss of

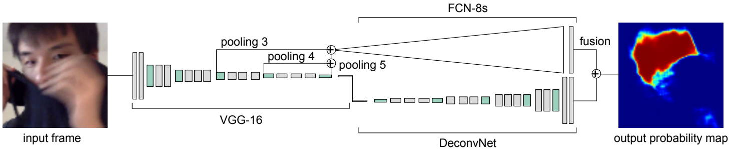


Fig. 2: ConvNet architecture with two-stream deconvolution network.

spatial information. DeconvNet on the other hand, mirrors the convolution process with multiple series of unpooling, deconvolution, and rectification layers, and generates detailed segmentations at the cost of increased noise. Noh and collaborators [12] proposed to combine the outputs of both algorithms through averaging followed by a post-hoc segmentation refinement based on conditional random fields [57], but the computation is prohibitively intensive. Instead, we develop an efficient network with shared convolution layers to reduce the number of parameters and operations, but split the deconvolution part into a two-stream architecture to benefit from the advantages of both networks. The output probability map resulting from a bilinear interpolation and mirrored deconvolution network are then concatenated before a final convolutional layer merges them into a single high-fidelity output map. We then use a standard graph cut algorithm [54] to convert the probability map into a clean binary facial mask and upsample to the resolution of the original input image via bilinear interpolation.

Architecture. Our segmentation network consists of a single convolution network connected to two different deconvolution networks, DeconvNet and an 8 pixel stride FCN-8s as shown in Figure 2. The network is based on a 16 layer VGG architecture and pre-trained on the PASCAL VOC 2012 data set with 20 object categories [13]. More specifically, VGG has 13 layers of convolutions and rectified linear units (ReLU), 5 max pooling layers, two fully connected layers, and one classification layer. DeconvNet mirrors the convolutional network to generate a probability map with the same resolution as the input, by applying upsampling operations (deconvolution) and the inverse operation of pooling (unpooling). Even though deconvolution is fast, the runtime performance is blocked by the first fully connected layer which becomes the bottleneck of the segmentation pipeline. To enable real-time performance on a state-of-the-art GPU, we reduce the kernel size of the first fully connected layer from 7×7 to 4×4 pixels.

Further modifications to the FCN-8s are needed in order to connect the output of both DeconvNet and FCN deconvolution networks to the final convolutional layer. The output size of each deconvolution is controlled by zero padding, so that the size of each upsampled activation layer is aligned with the output of the previous pooling layer. While the original FCN uses the last fully connected layer as the coarsest prediction, we instead use the output of the last pooling layer, as the coarsest prediction in order to preserve spatial information like in DeconvNet. The obtained coarse prediction is then sequentially deconvoluted and fused with the output of pooling layer 4 and 3, and then a deconvolution layer upsamples the fused prediction to the input image size. Since our 2-class labeling problem is considerably less complex than multi-class ones, losing information from discarded layers would not really affect the segmentation accuracy. In the final layer, the output of both deconvolution networks are concatenated into a single matrix and we apply a 1 × 1 convolution to obtain a score map, followed by a softmax operation to produce the final fused probability map. In this way we can even learn blending weights between the two networks as convolution parameters, instead of a simple averaging of output maps as proposed by the separate treatment of [12]. Please refer to the supplemental materials for the detailed configuration of our proposed network.

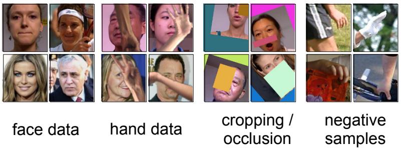


Fig. 3: Segmentation training data.

Training. For an effective facial segmentation, our convolutional neural network needs to be trained with large image datasets containing face samples and their corresponding ground truth binary masks. The faces should span a sufficiently wide range of shapes, appearance, and illumination conditions. We therefore collect 2927 images from the LFW face database [58] and 5094 images from the FaceWarehouse dataset [10]. While the LFW dataset already contains pre-labeled face segmentations, we segment those in FaceWarehouse using a custom semi-automatic tool. We use the available fitted face templates to estimate skin tones and perform a segmentation refinement using a graph cut algorithm [15]. Each sample is then manually inspected and corrected using additional seeds to ensure that occlusions such as hair and other accessories are properly handled.

To prevent overfitting, we augment our dataset with additional 82, 770 images using random perturbations of translation, rotation, and scale. The data consist of mostly photographs with a large variety of faces in different head poses, expressions, and under different lightings. Occlusions through hair, hands, and other objects are typically avoided. We therefore generate additional 82, 770 samples based on random sized and uniformly colored rectangles on top of each face sample to increase the robustness to partial occlusions (see Figure 3).

Skin toned objects such as hands and arms are commonly observed during hand-to-face gesticulations but are particularly challenging to segment due to similar colors as the face and fine structures such as fingers. We further augment the training dataset of our convolutional neural network with composited hands on top of the original 8021 face images. We first captured and manually segmented 1092 hand images of different skin tones, as well as under different lighting conditions and poses. We then synthesized these hand images on top of the original face images, which yields 41380 additional trainining samples using the same perturbation strategy. In total, 132, 426 images were generated to train our network. Our data-augmentation strategy can effectively train the segmentation network and avoid overfitting, even though only limited amount of ground truth data is available.

We initialize the training using pre-trained weights [13] except for the first fully connected layer of the convolution network, since its kernel size is modified for our real-time purposes. Thus, the first fully connected layers and deconvolution layers are initialized with zero-mean Gaussians. The loss function is the sum of softmax functions applied to the output maps of DeconvNet, FCN, and their score maps. The weights of each softmax function is set to 0.5, 0.5, and 1.0 respectively, and the loss functions are minimized via stochastic gradient descent (SGD) with momentum for stable convergence. Notice that by only using the fused score map of DeconvNet and FCN for the loss function, only the DeconvNet model is trained and not FCN. We set 0.01, 0.9, and 0.0005 as the learning rate, momentum, and weight decay, respectively. Our training takes 9 hours using 50,000 SGD iterations on our machines.

We further fine-tune the trained segmentation by adding negative samples (containing no faces) based on hand, arm, and background images to a random subset of the training data so that the amount of negative samples is equivalent to positive ones. In particular, the public datasets contain images that are both indoor and outdoors. Similar techniques for negative data augmentation has been used previously to improve the accuracy of weak supervision-based classifiers [59,60]. We use 4699 hand images that contain no faces from the Oxford hand dataset [61], and further perturb them with random translation and scalings. This fine-tuning with negative samples uses the same loss function and training parameters (momentum, weight decay, and loss weight) as with the training using positive data, but with initial learning rate of 0.001. Converges is reached after $10,000$ SGD iterations with an additional 1.5 hours of computation.

Segmentation Refinement. We convert the 128×128 pixel probability map of the convolutional neural network to a binary mask using a standard graph cut algorithm [15]. Even though our facial segmentation is reliable and accurate, a graph cut-based segmentation refinement can purge minor artifacts such as small 'uncertainty' holes at boundaries, which can still appear for challenging cases such as (extreme occlusions, motion blur, etc.). We optimize the following energy term between adjacent pixels i and j using the efficient GridCut [54] implementation:

$$\sum_{i} \theta_i(p_i) - \lambda \sum_{(i,j)} \theta_{i,j}. \tag{1}$$

The unary term $\theta_i(p_i)$ is determined by the facial probability map p_i, defined as $\theta_i(p_i) = -\log(p_i)$ for the sink and $\theta_i(p_i) = -\log(1.0 - p_i)$ for the source. The pairwise term $\theta_{i,j} = exp(-\frac{\|I_i - I_j\|^2}{2\sigma})$, where I is the pixel intensity, $\lambda = 10$, and $\sigma = 5$. The final binary mask is then bilinearly upsampled to the original cropped image resolution.

5 Facial Tracking

After facial segmentation, we capture the facial performance by regressing a 3D face model directly from the incoming RGB input frame. We adopt the state-of-the-art displaced dynamic expression (DDE) framework of [5] with the two-level boosted regression techniques of [34] and incorporate our facial segmentation masks into the regression and training process. More concretely, instead of computing the regression on face images with backgrounds and occlusions, where appearance can take huge variations, we only focus on segmented face regions to reduce the dimensionality of the problem. While the original DDE technique is reasonably robust for sufficiently large training datasets, we show that processing accurately segmented images significantly improves robustness and accuracy, since only facial apperance and lighting variations need to be considered. Even skin toned occlusions such as hands can be handled effectively by our method. We briefly summarize the DDE-based 3D facial regression and then describe how to explicitly incorporate facial segmentation masks.

DDE Regression. Our facial tracking is performed by regressing a facial shape displacement given the current input RGB image and an initial facial shape from the previous frame. Following the DDE model of [5], we represent a facial shape as a linear 3D blendshape model, $(\mathbf{b}_0, \mathbf{B})$, with global rigid head motion $(\mathbf{R}, \mathbf{t})$ and 2D residual displacements $\mathbf{D} = [\mathbf{d}_1 \dots \mathbf{d}_m]^T \in \mathbb{R}^{2m}$ of $m = 73$ facial landmark positions $\mathbf{P} = [\mathbf{p}_1 \cdots \mathbf{p}_m]^T \in \mathbb{R}^{2m}$ (eye contours, mouth, etc.). We obtain $\mathbf{P}$ through perspective projection of the 3D face with 2D offsets $\mathbf{D}$:

$$\mathbf{p}_i = \Pi_f(\mathbf{R} \cdot (\mathbf{b}_0^i + \mathbf{B}^i \mathbf{x}) + \mathbf{t}) + \mathbf{d}_i \quad , \tag{2}$$

where $\mathbf{b}_0^i$ is the 3D vertex location corresponding to the landmark $\mathbf{p}_i$ in the neutral face $\mathbf{b}_0$, $\mathbf{B} = [\mathbf{b}_1, ..., \mathbf{b}_n]$ the bases of expression blendshapes, $\mathbf{x} \in [0, 1]^n$ the $n = 46$ blendshape coefficients based on FACS [62]. Each neutral face and expression blendshape is also represented by a linear combination of 50 PCA bases of human identity shapes [9] with $[\mathbf{b}_0, \mathbf{B}] = C_r \times \mathbf{u}$, $\mathbf{u}$ the user-specific identity coefficients, and C_r the rank-3 core tensor obtained from the ZJU Face-Warehouse dataset [10]. We adopt a pinhole camera model, where the projection operator $\Pi_f : \mathbb{R}^3 \mapsto \mathbb{R}^2$ is specified by a focal length f. Thus, we can uniquely determine the 2D landmarks using the shape parameters $\mathbf{S} = \{\mathbf{R}, \mathbf{t}, \mathbf{x}, \mathbf{D}, \mathbf{u}, f\}$.

While the goal of the regression is to compute all parameters $\mathbf{S}$ given an input frame $\mathbf{I}$, we separate the optimization of the identity coefficients $\mathbf{u}$ and the focal length f from the rest, since they should be invariant over time. Therefore, the DDE regressor only updates the shape vector $\mathbf{Q} = [\mathbf{R}, \mathbf{t}, \mathbf{x}, \mathbf{D}]$ and $[\mathbf{u}, f]$ is computed only in specific key-frames and on a concurrent thread (see [5] for details). The two-level regressor structure consists of T sequential cascade regressors $\{R_t(\mathbf{I}, \mathbf{Q}_t)\}_{t=1}^T$ with updates $\delta\mathbf{Q}_{t+1}$ so that $\mathbf{Q}_{t+1} = \mathbf{Q}_t + \delta\mathbf{Q}_{t+1}$. Each of the weak regressors R_t classifies a set of randomly sampled feature points of $\mathbf{I}$ based on the corresponding pre-trained update vector $\delta\mathbf{Q}_{t+1}$. For each t, we sample new sets of 400 feature points via Gaussian distribution on the unit square. Notice that these points are represented as barycentric coordinates of a Delaunay triangulation of the mean of all 2D facial landmarks for improved robustness w.r.t. facial transformations. Each R_t consists of second layer of K primitive cascade regressors based on random ferns of size F (binary decision tree of depth F). Each fern regresses a weaker parameter update from a feature vector of F pixel intensity differences of feature point pairs from the 400 samples. The indices of feature point pairs are specified during training by maximizing the correlation to the ground truth regression residuals. The training process also determines the random thresholds and bin classification values of each fern.

At run-time, if a new expression or head pose is observed, we collect the resulting shape parameters $\hat{\mathbf{S}}$ as well as the landmarks $\hat{\mathbf{P}}$, and alternate the updates of the identity coefficients $\mathbf{u}$ and the focal length f by minimizing the offsets $\hat{\mathbf{D}}$ in Equation (2) for L collected key-frames until it converges as follows:

$$\underset{\mathbf{u},f}{\text{argmin}} \sum_{l=1}^{L}\sum_{i=1}^{m} \|\Pi_f(\hat{\mathbf{R}}_l \cdot (\mathbf{b}_0^i(\mathbf{u}) + \mathbf{B}^i(\mathbf{u}) \cdot \hat{\mathbf{x}}_l) + \hat{\mathbf{t}}_l) - \hat{\mathbf{p}}_{l,i}\|^2. \tag{3}$$

Training. The training process consists of constructing the ferns of the primitive regressors and specifying the F pairs of feature point indices based on a large database of facial images with corresponding ground truth facial shape parameters. We construct the ground truth parameters $\{\mathbf{S}_i^g\}_{i=1}^M$ from a set of images $\{\mathbf{I}_i\}_{i=1}^M$ and landmarks $\{\mathbf{P}_i\}_{i=1}^M$. Given landmarks $\mathbf{P}$, the parameters of the ground truth $\mathbf{S}^g$ are computed by minimizing the following objective function $\Theta(\mathrm{R}, \mathbf{t}, \mathbf{x}, \mathbf{u}, f)$:

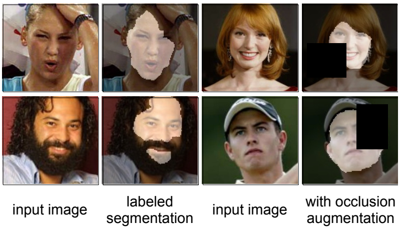


Fig. 4: Regression training data.

$$\Theta(\mathrm{R}, \mathbf{t}, \mathbf{x}, \mathbf{u}, f) = \sum_{i=1}^{m} \|\Pi_f(\mathbf{R} \cdot (\mathbf{b}_0^i(\mathbf{u}) + \mathbf{B}^i(\mathbf{u}) \cdot \mathbf{x}) + \mathbf{t}) - \mathbf{p}_i\|^2. \tag{4}$$

As in [5], we use 14, 460 labeled data from FaceWarehouse[10], LFW[58], and GTAV[63] and learn a mapping from an initial estimation $\mathbf{S}^*$ to the ground-truth parameters $\mathbf{S}^g$ given an input frame $\mathbf{I}$. An initial set of N shape parameters $\{\mathbf{S}_i^*\}_{i=1}^N$ are constructed by perturbing each training parameter in $\mathbf{S}$ within a predefined range. Let the suffix g denote the ground-truth value, suffix r a perturbed value.

We construct the training dataset $\{\mathbf{S}_i^* = [\mathbf{Q}_i^r, \mathbf{u}_i^g, f_i^g], \mathbf{S}_i^g = [\mathbf{Q}_i^g, \mathbf{u}_i^g, f_i^g], \mathbf{I}_i\}_{i=1}^N$ and perturb the shape vectors with random rotations, translations, blendshape coefficients as well as, identity coefficients $\mathbf{u}^r$ and the focal length f^r to improved robustness during training. Blendshapes are perturbed 15 times and the other parameters 5 times, resulting in a total of 506, 100 training data. The T cascade regressors $\{R_t(\mathbf{I}, \mathbf{Q}_t)\}_{t=1}^T$ then update $\mathbf{Q}$ so that the resulting vector $\mathbf{Q}_{t+1} = \mathbf{Q}_t + \delta\mathbf{Q}_{t+1}$ minimizes the residual to the ground truth $\mathbf{Q}^g$ among all training data N. Thus the regressor at stage t is trained as follows:

$$\delta\mathbf{Q}_{t+1} = \underset{R}{\text{argmin}} \sum_{i=1}^{N} \|\mathbf{Q}_i^g - (\mathbf{Q}_{i,t} + R_t(\mathbf{I}, \mathbf{Q}_{i,t}))\|_2^2. \tag{5}$$

Optimization. For both Equations 3 and 4, the blendshape and identity coefficients are solved using 3 iterations of non-linear least squares optimization with boundary constraints $\mathbf{x} \in [0, 1]^n$ using an L-BFGS-B solver [64] and the rigid motions $(\mathbf{R}, \mathbf{t})$ are obtained by interleaving iterative PnP optimization steps [65].

Segmentation-based Regression. To incorporate the facial mask $\mathbf{M}$ obtained from Section 4 into the regressors $R_t(\mathbf{I}, \mathbf{P}_t, \mathbf{M})$, we simply mark non-face pixels in $\mathbf{I}$ for both training and inference and prevent the regressors to sample

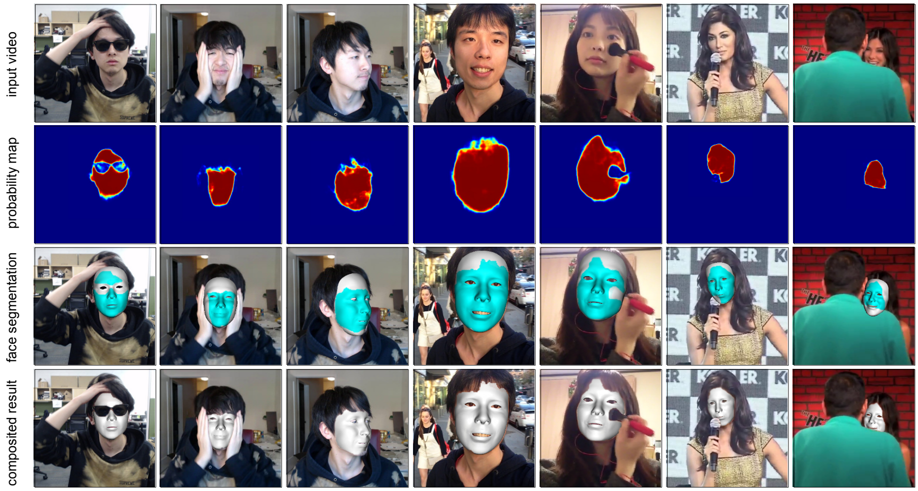


Fig. 5: Results. We visualize the input frame, the estimated probability map, the facial segmentation over the tracked template, and the composited result.

features in non-face region. To further enhance the tracking robustness under arbitrary occlusions, which is equivalent to incomplete views after the segmentation process, we augment the training data by randomly cropping out parts on the segmented face images (see Figure 4). For each of the 506, 100 training data sets, we include one additional cropped version with a rectangle centered randomly around the face region with Gaussian distribution and covering up to 80% of the face bounding box in width and height. Figure 8 and accompanied video shows that this occlusion augmentation significantly improves the robustness under various occlusions after data augmentation.

6 Results

As shown in Figure 5, we demonstrate successful facial segmentation and tracking on a wide range of examples with a variety of complex occlusions, including hair, hands, headwear, and props. Our convolutional network effectively predicts a dense probability map revealing face regions even when they are blocked by objects with similar skin tones such as hands. In most cases, the boundaries of the visibile face regions are correctly estimated. Even when only a small portion of the face is visibile we show that reliable 3D facial fitting is possible when processing input data with clean segmentations. In contrast to most RGB-D based solutions [7], our method works seamlessly in outdoor environments and with any type of video sources.

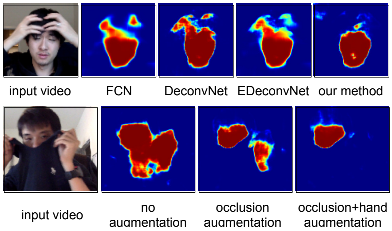


Fig. 6: Comparison of segmentation result based on different selection of neural network architectures.

Segmentation Evaluation and Comparison. We evaluate the accuracy of our segmentation technique on 437 color test images from the Caltech Occluded Faces in the Wild (COFW) dataset [48]. We use the commonly used intersection over union (IOU) metric between the predicted segmentations and the manually annotated ground truth masks provided by [66] in order to assess over and under-segmentations. We evaluate our proposed data augmentation strategy as well as the use of negative training samples in Figure 6 and show that the explicit use of hand compositings significantly improves the probability map accuracy during hand occlusions. We evalute the architecture of our network in Table 1 (left) and Figure 6 and compared our results with the state-of-the-art out of the box segmentation networks, FCN-8s[11], DeconvNet [12], and the naive ensemble of DeconvNet and FCN (EDeconvNet). Compared to FCN-8s and Deconvnet, the IOU of our method is improved by 12.7% and 1.4% respectively, but also contains much less noise as shown in Figure 6. While comparable to the performance of EDeconvNet, our method achieves nearly double the performance, which enables real-time capabilities (30 fps) on the latest GPU.

Network	mean IOU	FPS
FCN-8s	0.739	37.99
DeconvNet	0.821	44.31
EDeconvNet	0.835	20.45
Our Method	0.833	43.27

Method	mean IOU	global	ave(face)
Structured Forest [66]	-	0.839	0.886
RPP [67]	0.724	-	-
SAPM [47]	0.835	0.886	0.871
Our method	0.833	0.882	0.929
Our Method+GraphCut	0.839	0.887	0.927

Table 1: segmentation performance for different network structures (left) and state-of-the-art methods (right).

We compare in Table 1 (right), our deep learning-based approach against the current state-of-the-art in facial segmentation: (1) the structured forest technique [67], (2) the regional predictive power method (RPP) [66] and 3) segmentation-aware part model (SAPM) [51,47]. We measure the IOU and two additional metrics: global (the percentage of all pixels that are correctly classified) and ave(face) (the average recall of face pixels), since the structured forest work [67] uses these two metrics. We demonstrate superior performance to RPP (IOU: 0.833 vs 0.724) and structured forest (global: 0.882 vs 0.839, ave(face): 0.929 vs 0.886), and comparable result to SAPM (IOU: 0.833 vs 0.835, ave(face) 0.929 vs 0.871). Our method is significantly faster than SAPM which requires up to 30s per frame [51].

Tracking Evaluation and Comparison. In Figure 7, we highlight the robustness of our approach on extremely challenging cases. Our method can handle difficult lighting conditions, such as shadows and flashlights, as well as side views and facial hair. We further validate our data augmentation strategy during regression training and report quantitative comparisons with the current state-of-the-art method of Cao et al. [5] in Figure 8. Here, we produce an unoccluded face as ground truth and synthetically generated occluding box with increasing size. In

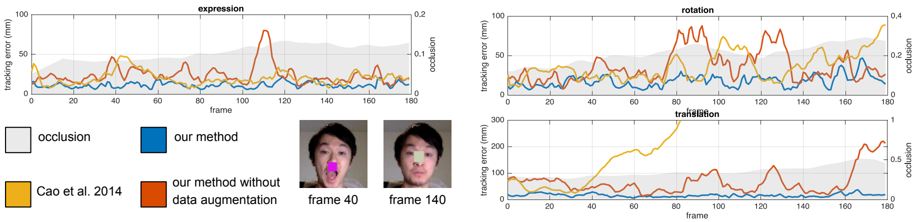


Fig. 8: Error evaluation on different tracking methods.

our experiment, we generated three sequences of 180 frames, covering a wide range of expressions, head rotations and translations.

We observe that our explicit semantic segmentation approach is critical to ensuring high tracking accuracy. While using the masked training dataset for regression significantly improves robustness, we show that additional performance can be achieved by augmenting this data with additional synthetic occlusions. Figure 9 shows how Cao et al.'s algorithm fails in the presence of large occlusions. Our method shows comparable occlusion-handling capabilities as the work of [7] who rely an RGB-D sensor as input. We demonstrate superior performance to a recent robust 2D landmark estimation method [48] when comparing the projected landmark positions. In particular, our method can handle larger occlusions and head rotations.

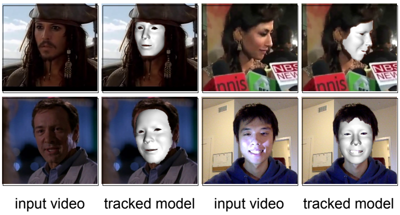


Fig. 7: Challenging tracking scenes.

Performance. Our tracking and segmentation stages run in parallel. The full facial tracking pipeline runs at 30 fps on a quad-core i7 2.8GHz Intel Core i7 with 16GB RAM and the segmentation is offloaded wirelessly to a quad-core i7 3.5GHz Intel Core i7 with 16GB RAM with an NVIDIA GTX Titan X GPU. During tracking, our system takes 18ms to regress the 3D face and 5ms to optimize the identity and the focal length. For segmentation, we measure the following timings: probability map computation 23ms, segmentation refinement 4ms, data transmission 1ms. run on the GPU, and the remaining implementation is multi-threaded on the CPU.

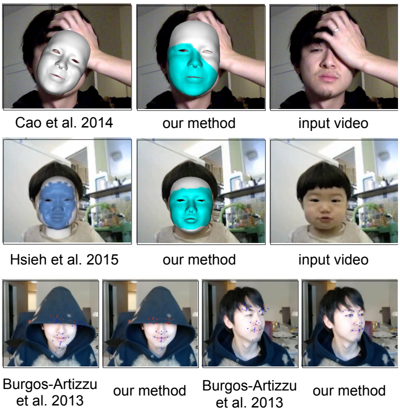


Fig. 9: Tracking comparison.

7 Conclusion

We demonstrate that real-time, accurate pixel-level facial segmentation is possible using only unconstrained RGB images with a deep learning approach. Our experiments confirm that a segmentation network with two-stream deconvolution network and shared convolution network is not only critical for extracting both the overall shape and fine-scale details effectively in real-time, but also presents the current state-of-the-art in face segmentation. We also found that a carefully designed data augmentation strategy effectively produces sufficiently large training datasets for the CNN to avoid overfitting, especially when only limited ground truth segmentations are available in public datasets. In particular, we demonstrate the first successful facial segmentations for skin-colored occlusions such as hands and arms using composited hand datasets on both positive and negative training samples. Significantly superior tracking accuracy and robustness to occlusion can be achieved by processing images with masked regions as input. Training the DDE regressor with images containing only facial regions and augmenting the dataset with synthetic occlusions ensures continuous tracking in the presence of challenging occlusions (e.g., hair and hands). Although we focus on 3D facial performance capture, we believe the key insight of this paper - reducing the dimensionality using semantic segmentation - is generally applicable to other vision problems beyond facial tracking and regression.

Limitations and Future Work. Since only limited training data is used, the resulting segmentation masks can still yield flickering boundaries. We wish to explore the use of a temporal information, as well as the modeling of domain-specific priors to better handle lighting variations. In addition to facial regions, we would also like to extend our ideas to segment other body parts to facilitate more complex compositing operations that include hands, bodies, and hair.

Acknowledgements

We would like to thank Joseph J. Lim, Qixing Huang, Duygu Ceylan, Lingyu Wei, Kyle Olszewski, Harry Shum, and Gary Bradski for the fruitful discussions and the proofreading. We also thank Rui Saito and Frances Chen for being our capture models. This research is supported in part by Adobe, Oculus & Facebook, Sony, Pelican Imaging, Panasonic, Embodee, Huawei, the Google Faculty Research Award, The Okawa Foundation Research Grant, the Office of Naval Research (ONR) / U.S. Navy, under award number N00014-15-1-2639, the Office of the Director of National Intelligence (ODNI), and Intelligence Advanced Research Projects Activity (IARPA), under contract number 2014-14071600010. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purpose notwithstanding any copyright annotation thereon.

References

1. Weise, T., Bouaziz, S., Li, H., Pauly, M.: Realtime performance-based facial animation. In: ACM Transactions on Graphics (TOG). Volume 30., ACM (2011) 77
2. Bouaziz, S., Wang, Y., Pauly, M.: Online modeling for realtime facial animation. ACM Trans. Graph. **32**(4) (2013) 40:1–40:10
3. Li, H., Yu, J., Ye, Y., Bregler, C.: Realtime facial animation with on-the-fly correctives. ACM Trans. Graph. **32**(4) (2013) 42
4. Cao, C., Weng, Y., Lin, S., Zhou, K.: 3D shape regression for real-time facial animation. ACM Trans. Graph. **32**(4) (2013) 41:1–41:10
5. Cao, C., Hou, Q., Zhou, K.: Displaced dynamic expression regression for real-time facial tracking and animation. ACM Transactions on Graphics (TOG) **33**(4) (2014) 43
6. Cao, C., Bradley, D., Zhou, K., Beeler, T.: Real-time high-fidelity facial performance capture. ACM Transactions on Graphics (TOG) **34**(4) (2015) 46
7. Hsieh, P.L., Ma, C., Yu, J., Li, H.: Unconstrained realtime facial performance capture. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 1675–1683
8. Faceshift: (2014) `http://www.faceshift.com/`.
9. Blanz, V., Vetter, T.: A morphable model for the synthesis of 3D faces. In: SIGGRAPH '99. (1999) 187–194
10. Cao, C., Weng, Y., Zhou, S., Tong, Y., Zhou, K.: Facewarehouse: a 3d facial expression database for visual computing. Visualization and Computer Graphics, IEEE Transactions on **20**(3) (2014) 413–425
11. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. CVPR (to appear) (November 2015)
12. Noh, H., Hong, S., Han, B.: Learning deconvolution network for semantic segmentation. In: Computer Vision (ICCV), 2015 IEEE International Conference on. (2015)
13. Chatfield, K., Simonyan, K., Vedaldi, A., Zisserman, A.: Return of the devil in the details: Delving deep into convolutional nets. In: British Machine Vision Conference. (2014)
14. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems. 2012
15. Rother, C., Kolmogorov, V., Blake, A.: "grabcut": Interactive foreground extraction using iterated graph cuts. In: ACM SIGGRAPH 2004 Papers. SIGGRAPH '04, New York, NY, USA, ACM (2004) 309–314
16. Pighin, F., Lewis, J.P.: Performance-driven facial animation. In: ACM SIGGRAPH 2006 Courses. SIGGRAPH '06 (2006)
17. Guenter, B., Grimm, C., Wood, D., Malvar, H., Pighin, F.: Making faces. In: SIGGRAPH '98. (1998) 55–66
18. Zhang, L., Snavely, N., Curless, B., Seitz, S.M.: Spacetime faces: High resolution capture for modeling and animation. ACM Trans. Graph. **23**(3) (2004) 548–558
19. Furukawa, Y., Ponce, J.: Dense 3D motion capture for human faces. In: CVPR. (2009) 1674–1681
20. Li, H., Adams, B., Guibas, L.J., Pauly, M.: Robust single-view geometry and motion reconstruction. ACM Trans. Graph. **28**(5) (2009) 175:1–175:10
21. Beeler, T., Hahn, F., Bradley, D., Bickel, B., Beardsley, P., Gotsman, C., Sumner, R.W., Gross, M.: High-quality passive facial performance capture using anchor frames. ACM Trans. Graph. **30** (2011) 75:1–75:10

22. Fyffe, G., Hawkins, T., Watts, C., Ma, W.C., Debevec, P.: Comprehensive facial performance capture. In: Computer Graphics Forum. Volume 30., Wiley Online Library (2011) 425–434
23. Bhat, K.S., Goldenthal, R., Ye, Y., Mallet, R., Koperwas, M.: High fidelity facial animation capture and retargeting with contours. In: SCA '13. (2013) 7–14
24. Fyffe, G., Jones, A., Alexander, O., Ichikari, R., Debevec, P.: Driving high-resolution facial scans with video performance capture. ACM Trans. Graph. **34**(1) (December 2014) 8:1–8:14
25. Li, H., Roivainen, P., Forcheimer, R.: 3-d motion estimation in model-based facial image coding. TPAMI **15**(6) (1993)
26. Bregler, C., Omohundro, S.: Surface learning with applications to lipreading. Advances in neural information processing systems (1994) 43–43
27. Black, M.J., Yacoob, Y.: Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motion. In: ICCV. (1995) 374–381
28. Essa, I., Basu, S., Darrell, T., Pentland, A.: Modeling, tracking and interactive animation of faces and heads using input from video. In: Proceedings of the Computer Animation. (1996) 68–79
29. Decarlo, D., Metaxas, D.: Optical flow constraints on deformable models with applications to face tracking. Int. J. Comput. Vision **38**(2) (2000) 99–127
30. Cootes, T.F., Edwards, G.J., Taylor, C.J.: Active appearance models. IEEE Transactions on Pattern Analysis & Machine Intelligence (6) (2001) 681–685
31. Cristinacce, D., Cootes, T.: Automatic feature localisation with constrained local models. Pattern Recogn. **41**(10) (2008) 3054–3067
32. Saragih, J.M., Lucey, S., Cohn, J.F.: Deformable model fitting by regularized landmark mean-shift. Int. J. Comput. Vision **91**(2) (2011) 200–215
33. Xiong, X., De la Torre, F.: Supervised descent method and its applications to face alignment. In: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, IEEE (2013) 532–539
34. Cao, X., Wei, Y., Wen, F., Sun, J.: Face alignment by explicit shape regression. International Journal of Computer Vision (2013)
35. Kazemi, V., Sullivan, J.: One millisecond face alignment with an ensemble of regression trees. In: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, IEEE (2014) 1867–1874
36. Ren, S., Cao, X., Wei, Y., Sun, J.: Face alignment at 3000 fps via regressing local binary features. In: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, IEEE (2014) 1685–1692
37. Weise, T., Li, H., Van Gool, L., Pauly, M.: Face/off: Live facial puppetry. In: Proceedings of the 2009 ACM SIGGRAPH/eurographics symposium on computer animation, ACM (2009) 7–16
38. Li, H., Weise, T., Pauly, M.: Example-based facial rigging. ACM Trans. Graph. **29**(4) (2010) 32:1–32:6
39. Pighin, F.H., Szeliski, R., Salesin, D.: Resynthesizing facial animation through 3D model-based tracking. In: ICCV. (1999) 143–150
40. Chuang, E., Bregler, C.: Performance driven facial animation using blendshape interpolation. Technical report, Stanford University (2002)
41. Chai, J., Xiao, J., Hodgins, J.: Vision-based control of 3D facial animation. In: SCA '03. (2003) 193–206
42. Suwajanakorn, S., Kemelmacher-Shlizerman, I., Seitz, S.: Total moving face reconstruction. In: Computer Vision – ECCV 2014. Volume 8692. Springer International Publishing (2014) 796–812
43. Garrido, P., Valgaerts, L., Wu, C., Theobalt, C.: Reconstructing detailed dynamic face geometry from monocular video. ACM Trans. Graph. **32**(6) (2013) 158

44. Shi, F., Wu, H.T., Tong, X., Chai, J.: Automatic acquisition of high-fidelity facial performances using monocular videos. ACM Transactions on Graphics (TOG) **33**(6) (2014) 222
45. Luo, P., Wang, X., Tang, X.: Hierarchical face parsing via deep learning. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, IEEE (2012) 2480–2487
46. Smith, B., Zhang, L., Brandt, J., Lin, Z., Yang, J.: Exemplar-based face parsing. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2013) 3484–3491
47. Ghiasi, G., Fowlkes, C.: Using segmentation to predict the absence of occluded parts. In: Proceedings of the British Machine Vision Conference (BMVC). (2015) 22.1–22.12
48. Burgos-Artizzu, X.P., Perona, P., Dollár, P.: Robust face landmark estimation under occlusion. In: Computer Vision (ICCV), 2013 IEEE International Conference on, IEEE (2013) 1513–1520
49. Gross, R., Matthews, I., Baker, S.: Active appearance models with occlusion. Image Vision Comput. **24**(6) (2006) 593–604
50. Ramanan, D.: Face detection, pose estimation, and landmark localization in the wild. In: CVPR. (2012) 2879–2886
51. Ghiasi, G., Fowlkes, C.C.: Occlusion coherence: Localizing occluded faces with a hierarchical deformable part model. In: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, IEEE (2014) 1899–1906
52. Yu, X., Lin, Z., Brandt, J., Metaxas, D.N.: Consensus of regression for occlusion-robust facial feature localization. In: ECCV. (2014) 105–118
53. Viola, P., Jones, M.: Robust real-time face detection. International Journal of Computer Vision **57**(2) (2004) 137–154
54. : Gridcut. `http://www.gridcut.com/`
55. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Semantic image segmentation with deep convolutional nets and fully connected crfs. arXiv preprint arXiv:1412.7062 (2014)
56. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR **abs/1409.1556** (2014)
57. Krähenbühl, P., Koltun, V.: Efficient inference in fully connected crfs with gaussian edge potentials. In Shawe-Taylor, J., Zemel, R.S., Bartlett, P.L., Pereira, F., Weinberger, K.Q., eds.: Advances in Neural Information Processing Systems 24. Curran Associates, Inc. (2011) 109–117
58. Huang, G.B., Ramesh, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst (October 2007)
59. Siva, P., Russell, C., Xiang, T.: In defence of negative mining for annotating weakly labelled data. In: Computer Vision–ECCV 2012. Springer (2012) 594–608
60. Song, H.O., Girshick, R., Jegelka, S., Mairal, J., Harchaoui, Z., Darrell, T.: On learning to localize objects with minimal supervision. arXiv preprint arXiv:1403.1024 (2014)
61. Mittal, A., Zisserman, A., Torr, P.H.S.: Hand detection using multiple proposals. In: British Machine Vision Conference. (2011)
62. Ekman, P., Friesen, W.: Facial action coding system: a technique for the measurement of facial movement. 1978. Consulting Psychologists, San Francisco
63. Tarrés, F., Rama, A.: Gtav face database. GVAP, UPC (2012)
64. Byrd, R.H., Lu, P., Nocedal, J., Zhu, C.: A limited memory algorithm for bound constrained optimization. SIAM J. Sci. Comput. **16**(5) (September 1995) 1190–1208

65. Lu, C.P., Hager, G.D., Mjolsness, E.: Fast and globally convergent pose estimation from video images. IEEE Trans. Pattern Anal. Mach. Intell. **22**(6) (June 2000) 610–622
66. Jia, X., Yang, H., Lin, A., Chan, K.P., Patras, I.: Structured semi-supervised forest for facial landmarks localization with face mask reasoning. In: Proc. Brit. Mach. Vis. Conf.(BMVA). (2014)
67. Yang, H., He, X., Jia, X., Patras, I.: Robust face alignment under occlusion via regional predictive power estimation. IEEE Transactions on Image Processing (2015)