A Discriminative Parts Based Model Approach for Fiducial Points Free and Shape Constrained Head Pose Normalisation In The Wild

Abhinav Dhall1  Karan Sikka2  Gwen Littlewort2  Roland Goecke3 1  Marian Bartlett2

1iHCC, Australian National University, Australia
2Machine Perception Laboratory, University of California San Diego
3Vision & Sensing Group, HCC Lab, University of Canberra, Australia
abhinav.dhall@anu.edu.au, gwen@mplab.ucsd.edu, roland.goecke@ieee.org, ksikka, mbartlett @ucsd.edu

Abstract

This paper proposes a method for parts-based view-invariant head pose normalisation, which works well even in difficult real-world conditions. Handling pose is a classical problem in facial analysis. Recently, parts-based models have shown promising performance for facial landmark points detection ‘in the wild’. Leveraging on the success of these models, the proposed data-driven regression framework computes a constrained normalised virtual frontal head pose. The response maps of a discriminatively trained part detector are used as texture information. These sparse texture maps are projected from non-frontal to frontal pose using block-wise structured regression. Finally, a facial kinematic shape constraint is achieved by applying a shape model. The advantages of the proposed approach are: a) no explicit dependence on the outputs of a facial parts detector and, thus, avoiding any error propagation owing to their failure; (b) the application of a shape prior on the reconstructed frontal maps provides an anatomically constrained facial shape; and c) modelling head pose as a mixture-of-parts model allows the framework to work without any prior pose information. Experiments are performed on the Multi-PIE and the ‘in the wild’ SFEW databases. The results demonstrate the effectiveness of the proposed method.

1. Introduction

In everyday situations and natural conversations, humans generally tend to move their head while speaking as part of non-verbal communication. This leads to several challenges, such as out-of-plane head rotations, (self-)occlusion and illumination variations. Facial landmark localisation and head pose handling play a vital role for facial analysis in fields such as human-computer interaction, biometrics, and affective computing, and have been active fields of research (e.g. [7, 20]). For instance, for face recognition and spontaneous facial expression analysis in real-world conditions, the head pose is normalised, to cancel the effect of head rotation [15], as a pre-processing step. The task of head pose normalisation (HPN) in particular aims at reconstructing the fiducial points for the input face in its frontal pose (referred to as virtual pose [15]) given a non-frontal face image (or fiducial points). Different from earlier approaches, this work proposes a view-invariant HPN method that does not require fiducial points as input in non-frontal pose and works directly with the input image (Figure 1).

Our approach employs the response maps generated by discriminatively trained facial part detectors. These confidence score maps are then normalised from non-frontal to frontal head pose using block-wise structure regression. A shape model is further applied on the virtual pose normalised confidence score maps to generate the virtual frontal landmark points. The entire approach is embedded in the Mixture of Pictorial Structures (MoPS) framework [20] to achieve robust performance on real-world images.

The contributions of the paper are as follows:
1. The proposed HPN approach is based on texture information generated from discriminative part detectors, unlike traditional approaches [1, 2, 16, 17], which are based on fiducial points.

2. The virtual frontal points generated by the proposed HPN methods are explicitly shape constrained. This overcomes the problem of standard regression based methods [1, 2, 17] where there is no implicit constraint on the shape of the object among the input and output data.

3. Previous methods [1, 2, 15, 16, 17] required head pose information for selecting a pose-specific regression model, which is certainly error-prone on real-world images. In contrast, the proposed method is head pose invariant.

Traditionally, pose-affected face analysis problems (recognition, expression analysis, etc.) can be broadly divided into two categories: a) top-down and b) bottom-up. In the former, the head pose is estimated first and then pose-specific classification models are used for inference [12, 13]. In the latter, the head pose is normalised first and then a frontal pose-specific classification model is used [2, 16, 17].

In one of the first works, Blanz et al. [4] proposed 3D Morphable Models for constructing 3D facial points from a single image. Asthana et al. [2] proposed a 3D HPN method using view-based Active Appearance Models (AAM) [8] and 3D model warping. The biggest drawback of these approaches is that the 3D models are computationally very expensive. 2D deformable model based approaches [8] overcome the computational problem. Facial landmark points are extracted using a 2D AAM and frontal pose points are computed using a linear regression model. However, such approaches only work well for expressionless faces.

Asthana et al. [1] proposed a regression-based method for generating faces at various poses. This method generated faces at different poses by learning a mapping from frontal to non-frontal facial landmark points. On similar lines Rudovic et al. [17] proposed a Gaussian Process Regression (GPR) [14] based HPN approach and compared different regression techniques. This work was then extended by coupling shape constraint with GPR (termed SC-GP) leading to performance improvements. The authors argued that without any explicit shape feature constraints, the normalised points may not adhere to the face shape.

A top-down approach was proposed by Huang et al. [12], which learned view-specific facial expression recognition (FER) models. During the inference step, a head pose estimator was used to select one of the view-specific FER models. Moore et al. [13] presented an extensive comparison of texture descriptors for multi-view FER. The proposed work in the paper is different from other HPN works, e.g. [12, 13, 17, 15], where the experiments were conducted on datasets captured in a lab-controlled environment only.

The shortcoming of approaches such as [1, 17] is that they require landmark points during inference. This is because robust facial landmark detection itself is an active research problem, particularly when dealing with real-world images, leading to errors in the results. This is in contrast to our parts-based approach, which does not require facial landmark points as input. On the other hand, approaches such as [13] require head pose information for selecting a pose-specific FER model. These approaches assume accurate results from the face detection and head pose estimation steps, which are both non-trivial tasks when working with real-world images. To remove the prerequisite of head pose estimation, Hu et al. [12] proposed to learn separate FER models for each pose. However, this further complicates the problem as increasing the number of non-frontal poses would also increase the number of models to be learnt and, thus, require more training data.

Further, [13] and [12] used hand crafted descriptors such as histogram of gradients (HOG) and local binary patterns (LBP). In contrast, part-based filters are learnt discriminatively for localising a particular part. The output from such filters is in spirit similar to discriminative mid-level representations or high-level features, which have been shown to outperform low-level features [18]. Another limitation of prior work [16, 17] is that fewer landmark points are used (39 in [16, 17]). This can be problematic, for example, in FACS-based facial action unit recognition such as AU20 (with no chin information). In contrast, our method generates a detailed 68-point annotation.

2. Mixture of Pictorial Structures

The MoPS framework [20] represents the parts of an object as a graph with $n$ vertices $V = \{v_1, \ldots, v_n\}$ and a set of edges $E$. Here, each edge $(v_i, v_j) \in E$ pair encodes the spatial relationship between parts $i$ and $j$. A face is represented as a tree graph here. Formally speaking, for a given image $I$, the MoPS framework computes a score for the configuration $L = \{l_i : i \in V\}$ of parts based on two models: an appearance model and a spatial prior model. These two models will be discussed now using the tree-based pictorial structures formulation similar to Zhu and Ramanan [20]. In particular, the formulation of [20] is followed.

The Appearance Model scores the confidence of a part-specific template $w_p$ applied to a location $l_i$. Here, $p$ is a view-specific mixture corresponding to a particular head pose. $\phi(I, l_i)$ is the histogram of oriented gradient descriptor [9] extracted from a location $l_i$. Thus, the appearance
model calculates a score for configuration $L$ and image $I$:

$$
App_p(I, L) = \sum_{i \in V_p} w_i^p \phi(I, l_i) \quad (1)
$$

The advantage of the part templates (detectors) in the appearance model is that less amount of data is required for training for each part detector. The response maps generated by these discriminative part detectors are sparse, which makes their reconstruction in the frontal view (for HPN) simpler. The Shape Model learns the kinematic constraints between each pair of parts. The shape model (as in [20]) is defined as:

$$
Shape_p(L) = \sum_{ij \in E_p} a_{ij}^p dx^2 + b_{ij}^p dx + c_{ij}^p dy^2 + d_{ij}^p dy \quad (2)
$$

Here, $dx$ and $dy$ represent the spatial distance between two parts. $a$, $b$, $c$, and $d$ are the parameters corresponding to the location and rigidity of a spring, respectively. From Eqs. 1 and 2, the scoring function $S$ is:

$$
Score(I, L, p) = App_p(I, L) + Shape_p(L) \quad (3)
$$

During the inference stage, the task is to maximise Eq. 3 over the configuration $L$ and mixture $p$ (which represents a pose). Therefore, if the pose of the face is known, then the inference is equivalent to finding the configuration $L^*$, which maximises the score for a given pose $p$:

$$
L^* = \max_L (Score(I, L, p)) \quad (4)
$$

When the pose is unknown, all models learnt for different values of $p$ are applied (Eq. 4) and the configuration specific to the highest scoring mixture is chosen as the facial parts locations.

3. Points Based Head Pose Normalisation

The points based HPN methods being discussed now are based on the idea of applying regression [1, 16, 17] over non-frontal points to obtain frontal points. For an image $I$ containing a non-frontal face, the fiducial point locations are computed using the parts-based model discussed in the previous section. For HPN, a mapping function (regressor) $F : L_p^i \rightarrow L_f^i$ is learnt that maps part locations in the non-frontal view to locations in the frontal view. $L_p^i$ and $L_f^i$ are the 2D coordinates of part $i$ in non-frontal and frontal pose, respectively. It should be highlighted that Rudovic et al. [17] also learned a similar mapping; however, during the test phase, manually defined landmark points were used as input. In contrast, in the proposed approach (Section 5), the part locations are computed automatically. Therefore, the results (Section 6) are closer to a real-world scenario and account for error due to face detection and facial parts localisation. Two different variants of points based HPN methods used in the experiments section (Section 6) are discussed below.

3.1. Part Wise Points (PWP) Based Normalisation

In PWP based HPN methods, frontal points are generated by regressing one point at a time using a point specific regression model. Based on univariate regression, $n$ models corresponding to each part (2D location) are learnt for each non-frontal pose. Thus, the total number of models learnt is $n \times P$, where $P$ is the number of non-frontal poses in the training data. The mapping function is then the regression function $R_l : L_p \rightarrow L_f$, i.e. the frontal location $L_f^i$ of each part is learnt from its corresponding non-frontal locations $L_p$. The major limitation of this method is that the outputs from different regression models are treated individually. As pointed out by [15], in such a case, there is no guarantee that the regressed part locations will adhere to the anatomical shape constraint of the face. The next model addresses this limitation.

3.2. All Parts Points (APP) Based Normalisation

The limitation of PWP HPN is overcome by learning a multi-variate regression model. In APP based HPN, frontal points are generated by learning a single regression model. That is, a function $R_l : L_p \rightarrow L_f$ that maps all parts in the non-frontal pose to all parts in the frontal pose is learnt. In the classic GPR framework [14], a multi-variate regression model is computed by mapping independent input points to single output dimension models. There is no explicit constraint modelling the relationship between the output dimensions. The model is made more robust by posing the APP based normalisation as a structured regression problem and using the twin Gaussian process regression (Twin-GPR) [5] framework. In the next section, Twin-GPR and its limitation when used for HPN are discussed.

4. Twin-GPR

The Twin-GPR framework models the relationship between the input and output variables. It uses Gaussian process priors on both covariances and responses, both multi-variate. The Kullback-Leibler divergence between the input and output data distributions, modelled as a Gaussian process, is minimised for capturing the correlation between the output dimensions. Bo et al. [5] proposed this method for regressing 2D human poses, as a structured regression problem, where the output dimensions are correlated by the human body kinematics. Similar to their problem, the intent in this paper is to reconstruct the facial points, where the points adhere to the anatomical face shape constraint. See Bo et al. [5] for details of the method.

Twin-GPR assumes that the input and output distributions are Gaussian. For real-world images, this assumption may not be satisfied due to the error introduced by the face alignment step [15]. This drawback is addressed in [15] by learning shape models based on ASM [7]. Shape parame-
Futhermore, a shape constraint is applied by exploiting properties of parts-based models. The mathematical formulation for cases with known pose is discussed in Section 5.1 and later extended to unknown poses (Section 5.2).

5.1. Pose-Specific Confidence Map Based HPN

Recall that Eqs. 1 and 2 are the appearance and shape components of the overall score (Eq. 3) optimised by the PS model. Given a non-frontal face image for pose $p$, parts-specific filters are applied via Eq. 1. This produces part-specific response maps (denoted by $A_{pp_i}$). For simplicity, the response of the appearance model for a particular part $i$ is denoted as a function $C_p^i$, which is defined as:

$$C_p^i = (I_i p)$$ (5)

The response $C_p^i$ will be a matrix of the size of the image $I_i$. Component $(x, y)$ of this matrix represents the probability of part $i$ being present at location $(x, y)$ in the image.

The task is to reconstruct the response map at the frontal pose for part $i$, referred to as $C_f^i$, from its response map $C_p^i$ at pose $p$. This is achieved by a structured regression model (Twin-GPR, Section 4) as discussed below. $C_f^i$ can be considered as a (synthesised) virtual frontal view response map for a part $i$. The motivation for using Twin-GPR is to maintain the relationship between neighbouring points being inferred in the frontal view response map.

The response maps $C_p^i$ is divided into blocks and a block-by-block Twin-GPR regression is learnt. This idea is motivated by the work of Biderman and Kalocsais [3], who discuss the importance of maintaining location information for facial parts when dealing with faces in a holistic manner.

Thus, each response map $C_p^i$ is first divided into $k$ equal sized non-overlapping blocks $B = B_1^p B_2^p \ldots B_k^p$ as shown in Figure 2 and a separate regression function is learnt for mapping each block. Thus, the big problem of mapping an entire confidence map is transformed into many smaller problems of mapping individual blocks with the aim of maintaining a structure. Non-overlapping blocks are preferred over a scanning window or overlapping blocks for their computational simplicity. Mathematically, during training we learn a set of models for each part $i$, denoted
5.2. Pose-Invariant Confidence Map Based HPN

As discussed in Section 5.1, when the head pose is unknown, the configuration $L^*$ of the highest scoring mixture $p$ is chosen as the best facial parts location. Based on this model, CM-HPN$_{PS}$ can be computed in a pose-invariant manner by simply enumerating over the $Score'(I, L, p)$ of each pose. Substituting $Score'$ (from Eq. 6) into Eq. 4 and maximising over all poses in the training data, the CM-HPN$_{PS}$ based $Virt_p$ based inference maximises:

$$L^*_p = \max_p \max_L \langle Score'(I, L, p) \rangle$$  \hspace{1cm} (7)

Here, $L^*_p$ is the highest scoring ‘virtual’ frontal head pose configuration. Basically, Algorithm 1 is computed for all poses $p$ in the training set and $L$ is the ‘virtual’ frontal head pose configuration generated with a regression model specific to a pose $p$.

6. Experiments

Our experiments on the Multi-PIE [11] dataset employed the same experimental protocol as Rudovic et al. [17]. This static facial expression dataset contains images from four pan angles (0°, -15°, -30°, -45°), with 200 images per pose. There are a total of 74 subjects and a five-fold cross validation over the samples was performed. The regression based HPN methods [1] and [17] were implemented for performance comparison and are referred to as PWP and APP (Section 3). The only difference with respect to the original implementations is that the facial points are located using the MoPS framework. Application of MoPS gives a performance advantage since it offers a better initialisation compared to the methods used in the original implementations of [1, 17]. [17] showed that the performance of Twin-GPR is better than GPR and SVM. Hence, Twin-GPR was used for learning PWP and APP. However for PWP, the output dimension is a single variate only (separate models are learnt for x and y positions).

Implementation details: The face area is located using the Viola-Jones face detector [19]. The faces are not aligned as a pre-processing step before the HPN step, as face alignment is a non-trivial problem for real-world images. Next, the detected face areas are rescaled to 320 x 240 pixels for consistency. The parameters for Twin-GPR are tuned empirically. The range of parameters experimented for the RBF kernel size is $[0.1 − 2]$ and $\lambda = [1.0e^{-1} − 1.0e^{-5}]$. The MoPS framework [20] is used and all experiments are based on independent models, since these were reported to be more accurate than shared models [20]. For the training details of the MoPS, see the original paper [20].

The performance of the points-based (PWP, APP) and confidence maps based HPN methods (CM-HPN$_{PS}$, CM-HPN$_{PI}$) was compared on the Multi-PIE database [11] using the error in the location of the reconstructed landmark.
points w.r.t. the frontal landmark points in the ground truth. Zhu and Ramanan [20] normalised the landmark location using the inter-ocular distance and the average of height and width of faces. Similar to [20], we used the Normalised Mean Square Error (NMSE), which describes the landmark localisation error (the L_2-distance between the virtual frontal points and the ground truth) normalised by the face size. This facilitates a fair comparison of the proposed methods with others in the future.

As discussed in Section 5, both CM-HPN_{PS} and CM-HPN_{PI} are computed block-by-block in a grid, whose configuration was chosen empirically. Different grid structures for non-overlapping blocks: [2 2 3 3 4 4 5 5] were compared. Table 1 summarises the performance comparison in terms of NMSE, showing that the models with 25 blocks performed the best achieving the lowest NMSE. This supports our hypothesis that dividing the maps into blocks reduces the complexity of the learnt model by maintaining a spatial constraint. As the number of blocks increase, the performance also increases until it saturates. For further comparison of the confidence maps based methods with the point-based ones, the highest performing grid configuration of 5 5 blocks was used.

Table 2 shows the performance for PWP, APP, CM-HPN_{PS} and CM-HPN_{PI}. The 5 5 blocks (i.e. 25 blocks in total in a confidence map) grid configuration was chosen for CM-HPN_{PS} and CM-HPN_{PI}. CM-HPN_{PS} has the smallest NMSE, performing the best. APP and CM-HPN_{PI} perform on par with each other (even though no prior head pose information is used in CM-HPN_{PI}). For the pose angle of 45°, the NMSE is fairly high for both points-based methods as compared to the proposed methods. This can be explained by the argument that as the head pose deviates away from the frontal view, computing facial points on the occluded side is error prone. We also observe that the reconstruction error is highest for PWP. This is primarily due to the lack of a relationship between the outputs of the different regressions models.

Ideally, the performance of CM-HPN_{PS} and CM-HPN_{PI} should be similar. However, CM-HPN_{PI} has a higher NMSE when the maximum score is achieved by an incorrect pose. For example, for a face with original head pose 45°, CM-HPN_{PI} will apply HPN with all models in R (for example, 45° 0° 30° 0°). If HPN with 30 model scores higher than the 45° model, then the method will assume that the face has a default head pose of 30° and will choose the corresponding incorrect reconstruction.

The expression wise NMSE is shown in Figure 3. For the Surprise expression, the error is large for all methods except CM-HPN_{PS}, which performs consistently best across all expressions. The biggest variation is for the points-based methods PWP and APP, which are based on [1, 15].

SFEW: To test the performance of CM-HPN_{PS} and CM-HPN_{PI} on ‘in the wild data’, the SFEW database [10] containing a set of video frames depicting facial expressions from movies, was used. Qualitative, visual comparison was performed as there is no frontal ground truth for SFEW unlike for Multi-PIE. We conducted a user survey (discussed below) for performance comparison on the HPN images produced by different methods. SFEW images, for which the pose was roughly similar to the Multi-PIE training set’s pose range, were chosen manually. In this analysis, the performance of a HPN method was tested for: a) images ‘in the wild’, b) unseen pose, and c) unseen expressions. We employed the HPN models that were trained on the Multi-PIE data for obtaining the virtual fiducial points for SFEW images. This reflects the power of discriminative part detectors, which, even with limited amounts of training data (Multi-PIE), are partially immune to attributes such as identity and illumination. This enables their use on unseen ‘in the wild data’ (e.g. SFEW).

Figure 4 shows the performance of the proposed methods on SFEW images. Results of point-based regression
methods are shown in columns 3 and 4, while those for the confidence maps based regression methods are shown in columns 5 and 6. It is evident from these images that the reconstruction of the overall shape for the PWP method is not as accurate as the other methods and that APP is not able to reconstruct the mouth correctly in some cases. Among all four methods, CM-HPN PS generally performs the best, but is unable to reconstruct eyes clearly in some cases. This can be addressed by employing denser grids. Note that the initialisation for the confidence score based methods is done by the Viola-Jones face detector. If a more accurate face detector such as MoPS itself is used, the reconstruction quality is expected to improve. It is also interesting to note that the jaw line of the reconstructed faces for the outputs of CM-HPN PS and CM-HPN PI shows a high degree of similarity to the jaw line shape of the subjects in the corresponding non-frontal images as compared to the output of APP, where the jaw line seems to be ‘averaged out’.

It is worth noting that [1, 17, 15] either used manually defined points or AAM. In contrast, the performance of APP can be attributed to robust landmark detection by the MoPS framework. As discussed earlier (Section 4), [15] proposed SC-GP to apply a shape constraint to overcome the problems arising due to inaccurate facial landmarks detection while regressing using Twin-GPR. Therefore, the performance of points-based methods can benefit from using a MoPS model.

In the last row of Figure 4, the reconstruction of the CM-HPN PI method is not accurate for the eyebrows. On further investigation, it is found that these errors are due to the error induced by the regression method, when the score of a non-frontal model’s frontal reconstruction is higher than the original non-frontal model’s reconstruction score. This can be corrected by applying efficient normalisation (for example, setting the mean to 0 and variance to 1) to data before regression. Twin-GPR is a generic structured regression model, the performance of the framework can be improved by using the class of structured SVM regression algorithms, which are problem specific. Further, based on the part sharing formulation, the method can be easily extended to continuous pose normalisation by sharing regression models among parts in neighbouring poses.1

A user survey was performed on SFEW, where 15 subjects were asked to rate the expression preserving ability of HPN for the 4 methods (PWP, APP, CM-HPN PS and CM-HPN PI) on a scale of 1 (poor) - 5 (excellent). CM-HPN PS and CM-HPN PI achieve mean values of 3.2 and 2.7, and standard deviations of 1.2 and 1.1, respectively. This is better than the ratings of PWP and APP, whose mean values are 1.9 and 2.5, and standard deviations of 1.1 and 1.2, respectively. Performing an ANOVA confirms that the result is statistically significant with $p < 0.0001$.

7. Conclusions

In this paper, we propose a new HPN method called Confidence Map based HPN. The method is based on confidence maps generated from parts based detectors and is embedded in the mixture of pictorial structure framework. The proposed method has no explicit dependency on facial parts location, thus making it suitable for images in real-world conditions. We also propose the use of a shape prior on reconstructed maps by applying a facial shape constraint. Further, enumerating over different poses allows our algorithm to work without any prior head pose information.

The results on the Multi-PIE database show the effectiveness of our methods in comparison to other state-of-the-art points based methods. We also show the generalisation capability of our algorithm using qualitative experiments on an ‘in the wild’ database (SFEW) by using pre-trained models from the Multi-PIE database. It is important to note that the images in SFEW are taken in more varied environments as compared to the laboratory-controlled environment in the Multi-PIE dataset.

The points-based approaches only provide geometric features, which are not appropriate for problems such as micro-expression and facial action unit analysis. Our methods provide both geometric and texture information (pose normalised response maps). Therefore, as part of future work, we will extend and experiment with the texture descriptors obtained as part of HPN.

References


1As continuous pose normalisation is not the focus of this paper, it is only briefly described in the supplementary material.
Figure 4. Performance comparison on selected SFEW images of PWP [1, 17], APP [15] and the proposed CM-HPN<sub>PS</sub> and CM-HPN<sub>PI</sub> regression methods. Columns 3-6 show the reconstructed landmark points for the face (Column 2).

<table>
<thead>
<tr>
<th>Original Image</th>
<th>Face Blob</th>
<th>PWP</th>
<th>APP</th>
<th>CM-HPN&lt;sub&gt;PS&lt;/sub&gt;</th>
<th>CM-HPN&lt;sub&gt;PI&lt;/sub&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Original Image" /></td>
<td><img src="image2" alt="Face Blob" /></td>
<td><img src="image3" alt="PWP" /></td>
<td><img src="image4" alt="APP" /></td>
<td><img src="image5" alt="CM-HPN&lt;sub&gt;PS&lt;/sub&gt;" /></td>
<td><img src="image6" alt="CM-HPN&lt;sub&gt;PI&lt;/sub&gt;" /></td>
</tr>
<tr>
<td><img src="image7" alt="Original Image" /></td>
<td><img src="image8" alt="Face Blob" /></td>
<td><img src="image9" alt="PWP" /></td>
<td><img src="image10" alt="APP" /></td>
<td><img src="image11" alt="CM-HPN&lt;sub&gt;PS&lt;/sub&gt;" /></td>
<td><img src="image12" alt="CM-HPN&lt;sub&gt;PI&lt;/sub&gt;" /></td>
</tr>
<tr>
<td><img src="image13" alt="Original Image" /></td>
<td><img src="image14" alt="Face Blob" /></td>
<td><img src="image15" alt="PWP" /></td>
<td><img src="image16" alt="APP" /></td>
<td><img src="image17" alt="CM-HPN&lt;sub&gt;PS&lt;/sub&gt;" /></td>
<td><img src="image18" alt="CM-HPN&lt;sub&gt;PI&lt;/sub&gt;" /></td>
</tr>
<tr>
<td><img src="image19" alt="Original Image" /></td>
<td><img src="image20" alt="Face Blob" /></td>
<td><img src="image21" alt="PWP" /></td>
<td><img src="image22" alt="APP" /></td>
<td><img src="image23" alt="CM-HPN&lt;sub&gt;PS&lt;/sub&gt;" /></td>
<td><img src="image24" alt="CM-HPN&lt;sub&gt;PI&lt;/sub&gt;" /></td>
</tr>
</tbody>
</table>