Three-D Wide Faces (3DWF): Facial Landmark Detection and 3D Reconstruction over a New RGB-D Multi-Camera Dataset

Quintana, M.; Karaoglu, S.; Alvarez, F.; Menendez, J.M.; Gevers, T.

DOI
10.3390/s19051103

Publication date
2019

Document Version
Final published version

Published in
Sensors

License
CC BY

Citation for published version (APA):
Abstract: Latest advances of deep learning paradigm and 3D imaging systems have raised the necessity for more complete datasets that allow exploitation of facial features such as pose, gender or age. In our work, we propose a new facial dataset collected with an innovative RGB–D multi-camera setup whose optimization is presented and validated. 3DWF includes 3D raw and registered data collection for 92 persons from low-cost RGB–D sensing devices to commercial scanners with great accuracy. 3DWF provides a complete dataset with relevant and accurate visual information for different tasks related to facial properties such as face tracking or 3D face reconstruction by means of annotated density normalized 2K clouds and RGB–D streams. In addition, we validate the reliability of our proposal by an original data augmentation method from a massive set of face meshes for facial landmark detection in 2D domain, and by head pose classification through common Machine Learning techniques directed towards proving alignment of collected data.

Keywords: face landmark detection; 3D face modelling; head pose classification; 3D data collection; deep learning

1. Introduction

Recent advances in computer vision and machine learning have directed the research community towards building large collections of annotated data in order to increase the performance of different application domains. In an analogous manner, 3D imaging has arrived by the introduction of low-cost devices and the development of efficient 3D reconstruction algorithms. In this paper, we focus on 3D facial imaging. Facial attributes range from simple demographic information such as gender, age, or ethnicity, to the physical characteristics of a face such as nose size, mouth shape, or eyebrow thickness, and even to environmental aspects such as lighting conditions, facial expression, and image quality [1].

The first attempt to collect facial data to study faces is through the introduction of Multi-PIE database [2], where the facial appearances vary significantly by a number of factors such as identity, illumination, pose, and expression. They built a setup with 13 cameras located at head height and spaced over 15 intervals. Multi-PIE database contains 337 subjects, imaged under 15 view points and 19 illumination conditions. This work is the starting point of the benchmark 300-W [3] that establishes the main metrics and standards to normalize the evaluation of the different methods for facial landmark analysis. One of the standards established by 300-W is the annotation of facial landmarks.
More recently, 2D traditional approaches for facial analysis have been using low-cost depth sensing devices by providing 3D modelling data sources. An important topic today is 3D face modeling employing RGB–D sensing devices. Traditional works mostly focus on providing accurate 3D face models. Recently, it was demonstrated that laser scan quality face models can be reconstructed from RGB–D sequences [4,5].

Head pose estimation is another important but challenging task for face analysis. Head pose provides important meta-information about communicative gestures, salient regions in a scene based on focus of attention, and can be used in surveillance environments to perform behaviour analysis [6,7].

It is known that deep learning methods are data hungry. Hence, there exists a data-demanding problem of deep networks, and tedious data annotation procedures related. This data problem can be solved by generating synthetic data [8,9]. In this paper, we have employed 3D meshes of 300 faces of subjects from 3DU dataset that allows for a proper augmentation method whose parameters are Euler angles [10] (pitch, yaw and roll). With this method, we only need to annotate every model once, but then we can generate thousands of augmented images for the different head poses.

The contributions of this work are:

1. **3DWF** We propose a multi-camera dataset containing visual facial features. We include streams from 600 to 1200 frames of RGB–D data from three cameras for 92 subjects and clouds normalized to 2 K points for ten poses proposed and their corresponding 3D landmarks projected. Demographic data such as age or gender is provided for every subject as well. Such a complete dataset in terms of the number of subjects and different imaging conditions is unique and the first of its kind.

2. An innovative **data augmentation method for facial landmark detection**. This method is based on a 3D(mesh)-2D projection implemented by raycasting [11].

3. **3D reconstruction workflow** adapted to facial properties and their normalization to provide meaningful features for cloud formats.

The paper is organized as follows:

1. **Section 2** outlines the techniques related to the work presented here.
2. **Section 3** presents the set-up for the acquisition of the proposed dataset.
3. **Section 4** introduces an innovative data augmentation for facial landmark detection.
4. **Section 5** describes a complete pipeline for 3D face modeling with a multi-camera RGB–D setup.
5. **Section 6** presents a validation method for the classification of the markers.
6. **Section 7** validates the captured data and evaluates the proposed methods for facial landmark detection.
7. The main contributions are discussed in **Section 8**.

2. Related Work

This section describes the current algorithms related to facial analysis that make use of computer vision techniques. We focus on the following facial analysis applications: 3D facial acquisition, facial landmarks detection, and head pose estimation.

2.1. **3D Facial Acquisition**

Recently, different approaches have been published to optimize acquisition systems to accurately represent 3D facial attributes. Ref. [12] collected 2000 2D facial images of 135 subjects as well as their 3D ground truth face scans. The authors proposed a dense face reconstruction method based on dense correspondence from every 2D image gathered to a collected neutral high resolution scan. Another hybrid solution (Florence Dataset) for reconstruction is presented in [13], where a complex capturing system (3dMD [14]) is used. The number of subjects (53) is also smaller than the one proposed here. UHDB31 [15] presented a follow-up of this work by increasing the number of poses and the number of
subjects resulting in a more complete dataset, but still costs of the set up are quite high and the number
of subjects are still under the ones captured in this work.

In our case, we provide neutral high resolution scans as well, but we believe that the impressive
results of recent works published, implementing deep learning for classification and segmentation
with normalized input clouds [16,17] motivate a new research line. Therefore, we postulate a new
challenge, and propose an initial reconstructed and normalized set to adopt this line for facial analysis.

The Pandora dataset, focusing on shoulder and head pose estimation, is introduced in [18] for
driving environments. However, the dataset only contains images from 20 subjects and they only have
one camera, and therefore it does not allow proper 3D reconstruction for extreme poses. Ref. [19]
proposes a 3D reference-free face modeling tested on a set of predefined poses. The authors perform
an initial data filtering process, and employ the face-pose to adapt the reconstruction. In our case,
we use 2D face detection projection, and afterward implement the proper 3D filtering techniques,
exploring information from 2D facial landmark detection in order to perform a more reliable 3D
face reconstruction. Other techniques are proposed by simply using a single RGB sensor, but in this
case they require either a 3D Morphable Model (3DMM) initially proposed by Blanz and Vetter [20],
this kind of method can be trapped in a local minimum and can not generate very accurate 3D geometry,
or a 2D reference frame and displacement measurement, as in [21].

2.2. Facial Landmark Detection

Research in this field is very proficient. Therefore, we concentrate on the three groups most
relevant to our work:

1. **Regression-Based Methods.** These methods directly learn a regression function from image
appearance (feature) to the target output (shape):

\[ M : F(I) \rightarrow x \in R^{2N} \]  

where \( M \) denotes the mapping from an image appearance feature \( F(I) \) to the shape \( x \), and \( F \)
is the feature extractor.

Ref. [22] proposed a two-level cascaded learning framework based on boosted regression. This method directly learns a vectorial output for all landmarks. Shape-indexed features are extracted from the whole image and fed into the regressor.

2. **Graphical Model-based Methods.** Graphical model-based methods mainly refer to
tree-structure-based methods and Markov Random Field (MRF) based methods.
Tree-structure-based methods take each facial feature point as a node and all points as a
tree. The locations of facial feature points can be optimally solved by dynamic programming.
Unlike the tree-structure that has no loops, MRF-based methods model the location of all points
by loops.

Zhu and Ramanan [23] proposed a unified model for face detection, head pose estimation and
landmark estimation. Their method is based on a mixture of trees, each of them corresponds
to one head pose view. These different trees share a pool of parts. Since tree-structure-based
methods only consider the local neighboring relation and neglect the global shape configuration,
they may easily lead to unreasonable facial shapes.

3. **Deep Learning-Based Methods.** Luo et al. [24] proposed a hierarchical face parsing method
based on deep learning. They recast the facial feature point localization problem as the process
of finding label maps. The proposed hierarchical framework consists of four layers performing
respectively the following tasks: face detector, facial parts detectors, facial component detectors
and facial component segmentation.

Sun et al. [25] proposed a three-level cascaded deep convolutional network framework for point
detection in a coarse-to-fine manner. It can achieve great accuracy, but this method needs to
model each point by a convolutional network that increases the complexity of the whole model.
Ref. [26] enhanced the detection by following a coarse-to-fine manner where coarse features inform finer features early in their formation, in such a way that finer features can make use of several layers of computation in deciding how to use coarse features. We selected this method to test the data augmentation method presented in Section 4.1 because of the novelty and efficiency of a deep net that combines convolution and max-pool layers to train faster than the summation baseline and yields more precise localization predictions.

Finally, the other selected solution to test the performance of the proposed augmentation method (Section 4.1) is [27] as they imply an evolution from previous models. As the proposed Tweaked Neural Network does not involve multiple part models, it is naturally hierarchical and requires no auxiliary labels beyond landmarks. They provide an analysis of representations produced at intermediate layers of a deep CNN trained for landmark detection, yielding good results at representing different head poses and (some) facial attributes. They inferred from previous analysis that the first fully connected layer already estimates rough head pose. With this information they can train pose specific landmark regressors.

### 2.3. Head Pose Estimation

Head pose estimation is a topic widely explored with applications such as autonomous driving, focus of attention modeling or emotion analysis. Fanelli et al. [28] introduced the first relevant method to solve this problem relying on depth sensing devices. Their proposal is based on random regression forests by formulating pose estimation as a regression problem. They synthesize a great amount of annotated training data using a statistical model of the human face. In an analogous way, Liu et al. [8] also propose a training method based on synthetic generated data. They use a Convolution Neural Network (CNN) to learn the most relevant features. To provide annotated head poses in the training process, they generate a realistic head pose dataset through rendering techniques. They fuse data from 37 subjects with differences in gender, age, race and expression. Some other lines of research, such as the one followed by [29], pose the problem as the classification of human gazing direction. We follow this approach in our work. It proposes as well deep learning techniques to fuse different low resolution sources of visual information that can be obtained from RGB–D devices. The authors encode depth information by adding two extra channels: surface normal azimuthal and surface normal elevation angle. Their learning stage is divided into two CNNs (RGB and depth inputs). The information learned by deep learning is employed to further fine-tune a regressor.

Analyzing all literature related to our work, we can conclude that our multi-camera RGB–D setup provides an affordable capturing system, able to perform 3D face reconstruction at extreme poses with a reasonable cost and deployment. In addition, facial landmark detection is already been explored extensively. Therefore, it is more suitable to provide a refinement of the 3D techniques being presented here. Head pose estimation is highly correlated with facial landmark detection (especially in 3D domain) and we believe with a good performance in facial landmarks head pose could easily be approached.

### 3. 3DWF Dataset

This dataset is captured by a system composed of 3 Asus Xtion depth cameras [30] in order to acquire multi-camera RGB–D information from 92 subjects by modifying their head poses steered by a sequence of markers. The subjects were asked to move their head continuously in a natural manner. To achieve a synchronous acquisition with three simultaneous devices, three independent USB buses are required, and synchronization among them has been implemented to provide a uniform acquisition. Synchronization among the devices is very critical since subjects are moving their head, and reducing delay between the cameras allows registration of the point clouds acquired. For that aim OpenNI 2 library [31] has been adopted by following the following procedure:

1. List of connected devices is gathered.
2. Every sensor is activated.
3. Data structures required to perform data flow are created.

First generation of RGB-D sensors are deployed due to their higher accuracy to perform 3D reconstruction of one category of objects proved in [32], and their feasibility to connect more than one device to the same computer. Between 600–1200 frames are recorded by each device for every subject. The proposed setup is displayed in Figure 1. Three RGB-D cameras can be observed in the Figure 1, the one in the middle will be named as frontal camera, and the other ones as side cameras. The number in the box represents the sequence of markers that the subjects were asked to follow (starting in box 1 and ending in box 10). Where \( W \) stands for width, \( H \) for height and \( D \) for depth, and origin is located at the frontal camera for \( W \) and \( D \), and the floor for \( H \). The proposed dataset contains the following sources:

1. Visual data
   (a) RGB and Depth data. This data has been continuously captured and is relevant for topics such as facial tracking or 3D face reconstruction.
   (b) RGB point clouds for ten markers. This data has been statically reconstructed with a target resolution of 2K, and it is very suitable for machine learning methods related to the tasks such as head pose or gaze estimation.
   (c) HD initial cloud. This data can be useful as reference cloud and has been captured with Faro Freestyle 3D Laser Scanner [33] whose 3D point accuracy is 0.5 mm and reconstructed by FARO Scene [34].

2. Subject data
   (a) Age
   (b) Gender

![Figure 1. Graphical description of the proposed scenario to capture the 3DWF dataset. Coordinates of the markers are expressed in meters.](image)

3.1. Set-Up Optimization

Optimization tests are mainly based on three parameters:
1. **Distance from the model to the frontal camera.** The manufacturer of the device recommends a distance in the range of 80–150 cm. Therefore, the tests are performed in this range. From Table A2, it can be derived that the highest number of points in the point cloud are obtained with distances of 80 cm. Analogously, the best visual appearance is gathered with that value.

2. **Light source.** Once the optimum distance from the model to the camera is calculated, the next step is to determine the parameters related to the LED light source employed whose main features are expressed in Table A1. Different tests are performed based on three parameters:

   (a) **Distance.** The tests performed for the distance were mainly based on visual appearance in the cloud obtained. For distances smaller than 200 cm appearance was too bright. We found that optimum distance should be set to 250 cm.

   (b) **Luminous flux.** We base our evaluation on the resolution of the pointcloud obtained for each camera. Results obtained can be noted in Table A3. It can be derived that as long as the luminous flux increases, the resolution of the point cloud decreases. Therefore, 250 Lumens (lm) (minimum provided by the manufacturer) is chosen.

   (c) **Orientation.** To optimize orientation of the light source at the proposed scenario, grayscale mean ($\hat{I}$) and Standard deviation ($\sigma$) values are evaluated. To this end, different angles between the light sources are explored with a luminous flux of 250 lm, but in this case, we also should consider the visual appearance. For that purpose, we have tested light sources pointing to three targets:

   i. Models
   ii. Frontal cameras
   iii. Side cameras

   Tests performed pointing to the models presented the worst visual results, even though different diffusion filters have been tested on the light source. Other values are shown in Table A4. The best visual results were obtained when light sources were pointing to the opposite side cameras. In this case, the mean and STD are replaced by the median ($\tilde{I}$) and Median Absolute Deviation (MAD), since the median and the mean own a notable difference. Also angles of the cameras are included in the table since they have a large influence on the results. Further, this parameter will be analyzed (separately) below.

3. **Camera orientation.** To test the best orientation of the cameras, different angles are used. All optimum parameters exposed previously are deployed in the scenario to test this parameter. The RGB–D devices chosen capture the scene affected by all parameters previously exposed, and therefore we believe it might be last parameter to be tested, and most critical since it determines the field of view. First tests are carried to determine the region of interest to be covered for the subjects involved in the experiment. The minimum angle required to completely cover the face of the subjects is $30^\circ$ and as long as the angles among the cameras are increased, surface covered increases as well. Finally, grayscale values obtained by the RGB sensor are evaluated to capture a similar range of color intensities for the faces. Results are shown in Table A5. It can be derived that as long as the angle among the cameras is increased, the difference between mean and the STD is also increased. Therefore, we can conclude that the optimum angle between the cameras is $30^\circ$.

3.2. **Subjects Description**

   For this dataset, age and gender are also registered for all the subjects. Statistics of those features are shown in Figure 2.

   We can observe that most of the population is located between 20 and 40 years old due to the fact that the dataset has been recorded in a university, but the dataset covers a wide range of ages. It is also noticeable that gender is a little unbalanced, however, looking to the specific graphs of age ranges for every gender it can be observed that age of females is more balanced than the age of males.
4. Facial Landmark Detection

This section proposes a new data augmentation method from 3D meshes to 2D images and analyzes its influence on two state of the art deep learning facial landmark detection methods.

4.1. Data Augmentation

With the data augmentation method proposed for face landmark detection we wanted to prove a possible application for the dataset proposed (3DWF). The dataset provided by 3DUniversum is captured by a rotating structured sensor device in order to reconstruct 3D models of 300 subjects. Rotation is performed by an analogous device to the one presented in [35], however, in this case projector is not required, and structured light sensor is combined with common RGB sensor to perform the capture. This dataset was gathered by performing a massive data collection which allowed to collect a larger number of subjects, in spite of collecting less facial attributes such as pose, gender of age from every one of them, 3D reconstruction of gathered data is outside of the scope of this work. In our case, we are directed towards a deep learning method to extract valuable features from meshes already processed. To this end, we used raycasting [11] to perform the projection of the 3D mesh to a 2D image. By implementing this technique, we moved the viewing plane in front of the pinhole to remove the inversion. A graphical explanation is shown in Figure 3. If an object point is at distance \( z_0 \) from the viewpoint, and has \( y \) coordinate \( y_0 \), then its projection \( y_p \) onto the viewplane is determined by the ratios of sides of similar triangles: \((0,0), (0,z_p), (y_p, z_p)\), and \((0,0), (0,z_0), (y_0, z_0)\). So we have:

\[
\frac{y_p}{z_p} = \frac{y_0}{z_0}
\]

The values of the viewpoint are based on the following parameters and values:

1. **Pitch.** \([-30^\circ, 30^\circ]\) Interval: 5°
2. **Yaw.** \([-30^\circ, 30^\circ]\) Interval: 5°
3. **Roll.** \([-30^\circ, 30^\circ]\) Interval: 5°
4. **Distance.** \([110 \text{ cm}, 160 \text{ cm}]\) Interval: 10 cm

Subjects are split in three datasets following the deep learning paradigm.

1. Training set: 245 subjects (81.67%).

**Figure 2.** Graphics showing the most important features of the subjects included in 3DWF dataset: age and gender.

---

4. Facial Landmark Detection

This section proposes a new data augmentation method from 3D meshes to 2D images and analyzes its influence on two state of the art deep learning facial landmark detection methods.

4.1. Data Augmentation

With the data augmentation method proposed for face landmark detection we wanted to prove a possible application for the dataset proposed (3DWF). The dataset provided by 3DUniversum is captured by a rotating structured sensor device in order to reconstruct 3D models of 300 subjects. Rotation is performed by an analogous device to the one presented in [35], however, in this case projector is not required, and structured light sensor is combined with common RGB sensor to perform the capture. This dataset was gathered by performing a massive data collection which allowed to collect a larger number of subjects, in spite of collecting less facial attributes such as pose, gender of age from every one of them, 3D reconstruction of gathered data is outside of the scope of this work. In our case, we are directed towards a deep learning method to extract valuable features from meshes already processed. To this end, we used raycasting [11] to perform the projection of the 3D mesh to a 2D image. By implementing this technique, we moved the viewing plane in front of the pinhole to remove the inversion. A graphical explanation is shown in Figure 3. If an object point is at distance \( z_0 \) from the viewpoint, and has \( y \) coordinate \( y_0 \), then its projection \( y_p \) onto the viewplane is determined by the ratios of sides of similar triangles: \((0,0), (0,z_p), (y_p, z_p)\), and \((0,0), (0,z_0), (y_0, z_0)\). So we have:

\[
\frac{y_p}{z_p} = \frac{y_0}{z_0}
\]

The values of the viewpoint are based on the following parameters and values:

1. **Pitch.** \([-30^\circ, 30^\circ]\) Interval: 5°
2. **Yaw.** \([-30^\circ, 30^\circ]\) Interval: 5°
3. **Roll.** \([-30^\circ, 30^\circ]\) Interval: 5°
4. **Distance.** \([110 \text{ cm}, 160 \text{ cm}]\) Interval: 10 cm

Subjects are split in three datasets following the deep learning paradigm.

1. Training set: 245 subjects (81.67%).
2. Validation set: 40 subjects (13.34%).
3. Testing set: 15 subjects (5%).

Figure 3. Raycasting geometry model with a plane and a pinhole. Extracted from [11] and reproduced with permission from Prof. House.

4.2. Deep Learning Architectures

4.2.1. VanillaCNN

The solution shown in [27] is selected as one suitable architecture to increase the performance of facial landmark detection for Annotated Facial Landmarks in the Wild Dataset (AFLW [36]) based on data augmentation method previously discussed. The architecture of this network includes mid-network features and implies a hierarchical learning. The main peculiarity of this network is the tweaking model oriented to two main processes:

1. It performs a specific clustering in the intermediate layers by a representation that discriminates between differently aligned faces. With that information, it trains pose specific landmark regressors.
2. The remaining weights from the first dense layer output are fine-tuned by selecting only the group of images classified in the same cluster with the features from the intermediate layers.

An absolute hyperbolic tangent is used as an activation function and Adam is used for training optimization [37]. L2 normalized by the inter-ocular distance is implemented as the network loss:

\[ \phi(P_i, \hat{P}_i) = \frac{\| P_i - \hat{P}_i \|^2_2}{\| \hat{P}_{i,1} - \hat{P}_{i,2} \|^2_2} \]  

(3)

where \( P_i \) is the 2xk vector of predicted coordinates for a training image, \( \hat{P}_i \) their ground truth locations, and \( \hat{P}_{i,1}, \hat{P}_{i,2} \) is the reference eye position.

4.2.2. Recombinator Networks (RCN)

Ref. [26] performs learning through using landmark independent feature maps. In this case, instead of performing specific learning, a more purely statistical approach is performed. The output of each branch is upsampled, then concatenated with the next level branch with one degree of finer resolution. Therefore, the main novelty is that branches pass more information to each other during training letting the network learn how to combine them non-linearly to maximize the log likelihood of the landmarks. It is only at the end of the Rth branch that feature maps are converted into a per-landmarks scoring representation by implementing a softmax.
All convolutional layers are followed by ReLU non-linearity except for the one right before the softmax. This architecture is trained globally using gradient backpropagation with an additional regularization term for the weights calculated through the next equation:

\[ L(W) = \frac{1}{N} \sum_{n=1}^{N} \sum_{k=1}^{K} -\log P(Y_k = y^{(n)}_k | X = x^{(n)}) + \lambda ||W||^2, \]  

where \( n \) is the number of samples, \( k \) is the number of landmarks, \( W \) represents the network parameters to minimize within regularization term to minimize and \( \lambda \) their weight.

In summary, we selected two architectures that alternate convolution and max-pooling layers, but whose nature is completely different. VanillaCNN presents four convolution layers and two dense layers. Dense layers are interlaid by a discrimination among the clusters previously learned from midnetwork features in a specific pose manner. RCN presents a bidirectional architecture with different branches including 3–4 convolution layers whose results are concatenated in the end of each branch to the inputs of the following one. VanillaCNN presents a descending size of filtering sizes along the network and RCN keeps it fixed.

5. 3D Reconstruction

This section validates the algorithm developed to present one point cloud for every subject for the markers located in the scenario that are graphically shown in Figure 1 in 3DWF Dataset. The steps performed are summarized in Figure 4.

![Figure 4. Block diagram of the different steps involved in 3D reconstruction of faces.](image)

5.1. Registration

Clouds obtained from RGB–D devices are registered by using a rigid body transformation [38]. We use an affine transformation [39]. Ten points are selected from every cloud (two by two matching). To obtain the transformation matrix, we built an homogeneous transformation, using the frontal point cloud as reference (\( Cl_{\text{Frontal}} \)):

\[ f(q_i) = Rp_i + t \]  

where \( R \) is the rotation matrix and \( t \) is the translation vector. Obtaining an origin matrix with a point in every row \( p_i = (x_i, y_i, z_i, 1) \) where \( i \in \{1, \ldots, 10\} \) from \( Cl_{\text{Right}} \) and \( Cl_{\text{Left}} \) and a target matrix with a point in every row \( q_i = (x_i, y_i, z_i, 1) \) where \( i \in \{1, \ldots, 10\} \) from \( Cl_{\text{Frontal}} \).

And to overcome accuracy errors in manual annotation and obtain optimum values for \( R_{\text{Left,Right}} \) and \( T_{\text{Left,Right}} \) we employed Random Sample Consensus (RANSAC) [40]. Then initial clouds (\( Cl_{\text{Left}}, Cl_{\text{Right}} \)) are transformed towards the reference frontal cloud (\( Cl_{\text{Left}}', Cl_{\text{Right}}' \)) and the resulting clouds are added two by two to obtain the complete cloud \( Cl_{\text{Total}} \). This summation task is shown in Figure 5.
5.2. Refinement

The second step for reconstruction is based on Iterative Closest Point (ICP) [41] algorithm. ICP is used to minimize difference between sets of geometrical points such as segments, triangles or parametric curves. In our work, we use the point-to-point approach. Metric distance between the origin cloud \((C_{\text{Left}} \text{ and } C_{\text{Right}})\) and target cloud \((C_{\text{Frontal}})\) is minimized by the following equation:

\[
i = \arg \min_i \| p_i - q_i \|^2,
\]

where \(p_i\) is the point belonging to the origin cloud \((C_{\text{Left}} \text{ and } C_{\text{Right}})\) and \(q_i\) is a point belonging to the target cloud \((C_{\text{Frontal}})\). Regarding rotation and translation matrix, the algorithm iterates over the minimum square distances by:

\[
R, t = \sum_{i=1}^{N} \arg \min_{R,T} \| (R p_i + t) - q_i \|^2,
\]

where \(N\) is the number of iterations fixed to 30 for our solution and we have also fixed the percentage of worst candidate removal to 90%. With this refinement \(C_{\text{Total}'}\) is obtained.

5.3. ROI

Once the whole cloud is built and refined, we use the Dlib face detector [42] on the RGB image from the frontal camera in order to determine the region of interest (ROI). In a similar way, we apply the face landmark detection based on VanillaCNN exposed in Section 4 obtaining the locations of facial features.
landmarks. To project the keypoints obtained from the neural network, we use Perspective Projection Model [43]. By applying the following equations:

\[ X_k = -\frac{Z_k}{f}(x_k - x_c + \delta_x), \quad Y_k = -\frac{Z_k}{f}(y_k - y_c + \delta_y), \]

where \( X_k, Y_k \) and \( Z_k \) are the projected coordinates in the cloud, \( x_c \) and \( y_c \) are the coordinates of the center of the 2D image, \( x_k \) and \( y_k \) are the input coordinates from the 2D image and \( \delta_x \) and \( \delta_y \) are the parameters to correct the distortion of the lens provided by the manufacturer. Obtaining 3D projection for the bounding box delimiters to project the cropped cloud obtained after refinement \( Cl_{Total'} \) into a cloud with mostly facial properties \( Cl_{F0} \).

In order to test the accuracy of \( Cl_{F0} \) we have considered the clouds gathered with Faro Freestyle 3D Laser Scanner (\( Cl_{HD} \)) as ground truth, and we have measured the average minimum distance from \( \forall pt_i \in Cl_{F0} \) to \( \forall pt'_i \in Cl_{HD} \) for every subject, obtaining as result distances in the range \([16 - 23] \) mm. In addition, we should consider:

1. Since the distance from the subject to the camera is below 1 m, the error of the depth sensor should be in the range \([5 - 15] \) mm according to the results exposed in [44].
2. The faces of the subjects are not rigid (although both captures have been performed on a neutral pose).

Therefore, the range measured as distance from \( Cl_{F0} \) to \( Cl_{HD} \) proves the accuracy of the 3D reconstruction performed.

5.4. Noise Filtering

In this section, an algorithm to filter \( Cl_{F0} \) is proposed to obtain reliable face clouds. The following features are proposed:

1. **Color.** Initially we need to delimit two areas:

   (a) 2D ROI’ to extract. We have employed facial landmarks detected by VanillaCNN through data augmentation procedure presented in Sections 4.2 and 4.1 respectively. In our setup we have detected five points: left eye (le), right eye (re), nose (n), left mouth (lm) and right mouth (rm). A new ROI (ROI’) is defined based on a bounding box with these detections:

   \[ \{ (x_{le}, y_{le}), (x_{re}, y_{re}), (x_{lm}, y_{lm}), (x_{rm}, y_{rm}) \} \]

   ROI’\text{RGB} intensities are transformed to a more uniform color space: CIELAB [45]. Components values of the two intensities samples used for thresholding are calculated in the following manner following a normal distribution:

   \[
   \begin{align*}
   th_{Lab} &= (\hat{L}_{ROI'} \pm w\sigma_{L_{ROI'}}, \hat{a}_{ROI'} \pm w\sigma_{a_{ROI'}}, \hat{b}_{ROI'} \pm w\sigma_{b_{ROI'}}) \\
   \end{align*}
   \]

   where \( \hat{L}_{ROI'} \) and \( \sigma_{L_{ROI'}} \) are the mean and standard deviation of L component from CIELAB color space for the new ROI defined. Analogously for \( \hat{a}_{ROI'} \), \( \sigma_{a_{ROI'}} \), \( \hat{b}_{ROI'} \) and \( \sigma_{b_{ROI'}} \). And \( w \) is fixed to 0.75 in our implementation.

   (b) 3D Contour \( Cl_{F0} \). In this case we have defined two margins for width and height from \( Cl_{F0} \) to filter farther points to the cloud centroid by applying CIEDE2000 \( \forall pt_i \in Cl_{F0} \):

   \[
   \begin{align*}
   & if \Delta E_{00}^s \{ (L_{ROI'}, \hat{a}_{ROI'}, \hat{b}_{ROI'}), (L_{pt_i}, a_{pt_i}, b_{pt_i}) \} < \\
   & \Delta E_{00}^s(th_{1_{Lab}}, th_{2_{Lab}}) \Rightarrow pt_i \in Cl_{FC} \end{align*}
   \]
where \( \Delta E^*_{00} \) is the metric used in CIEDE2000 and \( Cl_{f_{fc}} \) is the point cloud obtained after color filtering.

2. **Depth.** Mainly focused on noise introduced by depth sensors and outliers from color filtering. For that aim we have built a confidence interval based on normal distribution of \( Cl_{f_{fc}} \) 
\[
\bar{Z}_{Cl_{f_{fc}}} \pm w_Z \sigma_{Z_{Cl_{f_{fc}}}}.
\]
where \( w_Z \) is fixed to 2.25 in our implementation.

5.5. **Uniform Distribution**

To provide a reliable point cloud dataset, it is important that clouds have similar resolutions and that every part of the cloud is constant regarding point-space density. For that reason, \( Cl_{f_{fc}} \) is divided into four parts based on its width and height. A resolution of 2K points is proposed as target resolution. Therefore, every cloud part should have 2K/4 points. An implementation of voxel grid downsampling [46] based on a dynamic radius search is used. The voxel grid filter down-samples the data by taking a spatial average of the points in the cloud through employing rectangular areas that are known as voxels. The set of points that lie within the bounds of a voxel are assigned to that voxel and will be combined into one output point. With this final step \( Cl_{f_{fc}} \) is composed, and sample values for one subject are displayed in Figure 6. In an analogous way one sample for Marker 1 without texture mapping is shown in Figure 7.

![Figure 6](image1.png)

*Figure 6. 2D images extracted from the final face clouds proposed by this work.*

![Figure 7](image2.png)

*Figure 7. 2D images extracted from the final face clouds for Marker 1 without texture mapping proposed by this work.*

6. **Head Pose Classification**

This section describes the methods implemented to validate the alignment of head pose values of the data gathered in 3DFW dataset with markers located in the scene. We used visual information of subjects when they look at marker 1 (relaxed pose looking to the front) as reference for the other markers. Initial steps are analogous to the ones proposed in Section 5.3. In this case, we reversely used projection equations shown in (8) together with 2D Euclidean distance to gather the closest points included in \( Cl_{Total} \) to 2D facial landmarks detected by Vanilla CNN. In this way, a new set composed by 3D facial landmarks is obtained:
Rigid Motion

Initial transformations are performed by using the Least-Squares Rigid Motion by means of SVD \cite{47} from the set of 3D facial landmarks to obtain the corresponding rotation matrix. Let \( P = (p_1, p_2, ..., p_n) \) where \( p_i \) are 3D coordinates of facial landmarks for marker 1 \( \in \mathbb{R}^3 \) and \( Q = (q_1, q_2, ..., q_n) \) where \( q_i \) are 3D coordinates of facial landmarks for markers 2–10 \( \in \mathbb{R}^3 \) be our reference and target sets of data respectively. We are able to find a rigid transformation that optimally aligns the two sets in the least squares sense, i.e., assuming unity vector for translation matrix (subjects are static in the experiment proposed):

\[
R = \argmin_{R \in SO(3)} \sum_{i=1}^{n} w_i ||Rp_i - q_i||^2
\]  

(11)

By Restating the problem so that the translation would be zero, and simplifying the expression we can reformulate the problem:

\[
||Rp_i - q_i||^2 = tr(WY^T RX)
\]  

(12)

where \( W = \text{diag}(w_1, ..., w_n) \) is an \( nxn \) diagonal matrix with the weight \( w_i \) on diagonal entry \( i \), \( Y \) is the \( dxn \) matrix with \( y_i \) as its columns and \( X \) is the \( dxn \) matrix with \( x_i \) as its columns. \( tr \) is the trace of a square matrix (sum of the elements on the diagonal) and owes commutative property with respect to product. Therefore we are looking for a rotation \( R \) that maximizes \( tr(RXWY^T) \). Now we have denoted \( dxd \) covariance matrix \( S = XWy^T \). If we take Single Value Decomposition (SVD) of \( S \) such that \( S = U\Sigma V^T \) relying on the fact that that \( V, R \) and \( U \) are all orthogonal matrices, so \( V^T RU \) is also an orthogonal matrix and we can assume identity. Therefore we can calculate corresponding rotation matrix in the following way:

\[
R = VU^T
\]  

(13)

Given a rotation matrix \( R \), we can compute the Euler angles, \( \phi, \theta, \psi \) by equating each element in \( R \) with the corresponding element in the matrix product \( R_Z(\phi)R_Y(\theta)R_X(\psi) \). This results in nine equations that have been used to find the Euler angles.

7. Results

In this subsection, metrics used for evaluation and results obtained by them for the proposed architectures and datasets are presented.

7.1. Facial Landmark Detection

7.1.1. Training Details

Initially, we have defined training details for every dataset:

1. **3DU Dataset.** Number of viewpoints augmented to split dataset, number of subjects included per view and learning rate base for 3DU Dataset are expressed in Table 1. Subjects are randomized for every viewpoint based on splits presented in Section 4.1.

2. **AFLW.** Number of images to split dataset and learning rate base for AFLW Dataset are shown in Table 2.
Table 1. Parameters for training 3DU dataset.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Subjects</th>
<th>Learning Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train.</td>
<td>Val.</td>
<td>RCN</td>
</tr>
<tr>
<td>3DU</td>
<td>35</td>
<td>12</td>
</tr>
</tbody>
</table>

Table 2. Parameters for training AFLW dataset.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Images</th>
<th>Learning Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train.</td>
<td>Val.</td>
<td>Test</td>
</tr>
<tr>
<td>AFLW</td>
<td>9000</td>
<td>3000</td>
</tr>
</tbody>
</table>

7.1.2. Error Metric

The euclidean distance between the true and estimated landmark positions normalized by the distance between the eyes (interocular distance) is used:

$$
\epsilon = \frac{1}{KN} \sum_{n=1}^{N} \sum_{k=1}^{K} \sqrt{\left(\hat{w}_{k}^{n} - \hat{w}(n)\right)^2 + \left(\hat{h}_{k}^{n} - \hat{h}(n)\right)^2} / D(n)
$$

where $K$ is the number of landmarks (5 in our work), $N$ is the total number of images, $D(n)$ is the interocular distance in image $(n)$. $(w_{k}^{n}, h_{k}^{n})$ represent the true and $(\hat{w}_{k}^{n}, \hat{h}_{k}^{n})$ estimated coordinates for landmark $k$ in image $n$, respectively. Localization error is measured as a fraction of the inter-ocular distance, a measure invariant to the actual size of the images. We declare a point correctly detected if the pixel error is below 0.1 interocular distance.

7.1.3. Accuracy

The histogram in Figure 8 show that the error for the different combinations of networks, explained in Section 4.2 and datasets. It can be derived that the lowest error rate is by training RCN with the 3DU dataset.

Massive normalized 2D projections of this dataset learned in a bidirectional approach reduces the error. In addition, we mention that the network layers based on midnetwork features proposed by VanillaCNN achieve the worst results with the same training and testing data since this is an specific solution for another nature of data, but weights and bias pre-learned from 3DU dataset increases the performance of the algorithm, and helps to achieve the best results for AFLW after finetuning. In this case, massive data initialize properly midnetwork features so that the network can go beyond in global minimum target for common datasets of landmark detection such as AFLW. This procedure is shown, using the visualization of filters, in Tables 3 and 4 for last convolution and max-pooling layers, where brighter color intensities represent stronger activations. It can be derived that the initial training of 3DU provides blunter features at this stage of the learning due to triangulation procedure to gather mesh input data. It can be inferred as well that the fine-tuning process provides sharper features that increase the performance of the network.
Figure 8. Results obtained for both pipelines of face landmark detection with different combinations of datasets.

Table 3. Representation of the last layer of convolution from VainillaCNN.

<table>
<thead>
<tr>
<th></th>
<th>Conv 3DU</th>
<th>Conv 3DU + AFLW</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><img src="image1" alt="Image" /></td>
<td><img src="image2" alt="Image" /></td>
</tr>
</tbody>
</table>

Table 4. Representation of the last layer of max-pooling from VainillaCNN.

<table>
<thead>
<tr>
<th></th>
<th>Pool 3DU</th>
<th>Pool 3DU + AFLW</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><img src="image3" alt="Image" /></td>
<td><img src="image4" alt="Image" /></td>
</tr>
</tbody>
</table>

7.2. Head Pose Classification

To validate the proposed dataset, we assumed Euler angles calculated previously as head pose angles (pitch (φ), yaw (ψ) and roll (θ)). Three 2D projections of this data can we noted on Figure 9. It can be inferred that the distribution of the data is quite uniform.

Figure 9. Graphical plots of Euler Angles obtained for the different markers of 3DWF.
Accuracy

Two validation methods have been tested in order to classify the Euler angles calculated for the projection of the facial landmarks obtained by fine tuning the initial training of VanillaCNN with the data augmentation procedure exposed in Section 4.1 with the training of AFLW. The methods selected are Linear Discriminant Analysis (LDA [48]) and Gaussian Naive Bayes (GNB [49]). The aim of those classification methods is to validate the data capture, and the projection of the facial landmarks estimated to the point clouds gathered. The results and the main features of the proposed methods can be noticed on Table 5.

Table 5. Results for head pose classification.

<table>
<thead>
<tr>
<th>Method</th>
<th>Training Samples</th>
<th>Testing Samples</th>
<th>Training Accuracy</th>
<th>Testing Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>LDA</td>
<td>80%</td>
<td>20%</td>
<td>82%</td>
<td>83%</td>
</tr>
<tr>
<td>GNB</td>
<td>80%</td>
<td>20%</td>
<td>84%</td>
<td>84%</td>
</tr>
</tbody>
</table>

Confusion matrix for GNB classification, where rows correspond to ground truth markers (2–10) and columns to predicted markers in the same range, is shown in Figure 10. Results show promising values for a simple classification technique such as GNB. It can be noticed that those markers where the subjects are looking to one side of the scene (such as 2 or 5) are the most complex to predict, and those markers where the subjects are looking straight and modifying their pitch (such as 10) the simplest.

8. Conclusions

In this paper, we have presented an optimized multi-camera RGB–D system for facial properties to capture accurate and reliable data. In this scope we have performed a data collection including 92 people fulfilling the need of a 3D facial dataset able to exploit capabilities of deep learning paradigm in 3D scope. In addition, we provide a complete pipeline to process data collected and pose a challenge for Computer Vision and Machine Learning research community by annotating human characteristics such as age or gender. The collected RGB–D streams allow other related tasks such as face tracking or 3D reconstruction with a wide source of visual information that increase the performance of common acquisition systems for extreme head poses.

In this scope, we found facial landmark detection one of the main tasks where our work should contribute to research lines that project 3D information into a more feasible and less costly domain.
such as 2D. For that reason, we have proposed an innovative data augmentation method, tested and discussed its accuracy on two state-of-the-art deep learning solutions. We have trained and evaluated synthetic and visual imaging data on two complementary architectures, finding a combined solution that enhances results for a very common deep net architecture like Vanilla CNN.

Finally, the alignment of the path proposed to the subjects by ten markers is validated by implementing a geometric approach for head-pose through previously estimated features. The refinement of the learning techniques implemented for this task is one of the lines of research proposed for future work.

**Author Contributions:** Conceptualization, S.K. and T.G.; Formal analysis, M.Q., S.K. and J.M.M.; Funding acquisition, J.M.M.; Investigation, M.Q.; Methodology, J.M.M. and T.G.; Resources, F.A.; Software, M.Q.; Supervision, F.A.; Visualization, M.Q.; Writing—original draft, M.Q.; Writing—review & editing, S.K., F.A. and T.G.

**Funding:** This project has been partially funded by the European project AI4EU: “Advancing Europe through collaboration in AI”.

**Conflicts of Interest:** The authors declare that there is no conflict of interest regarding the publication of this article.

**Abbreviations**

The following abbreviations are used in this manuscript:

- **RGB-D**  Red Green Blue-Depth
- **3DWF**  3D Wide Faces
- **Multi-PIE**  Multi Pose Illumination Expression
- **3DU**  3DUniversum
- **3DMM**  3D Morphable Model
- **MRF**  Markov Random Field
- **CNN**  Convolutional Neural Network
- **USB**  Universal Serial Bus
- **LED**  Light-Emitting Diode
- **STD**  Standard Deviation
- **MAD**  Median Absolute Deviation
- **AFLW**  Annotated Facial Landmarks in the Wild Dataset
- **RCN**  Recombinator Networks
- **ReLU**  Rectifier Linear Unit
- **RANSAC**  Random Sample Consensus
- **ICP**  Iterative Closest Point
- **ROI**  Region Of Interest
- **CIELAB**  color space defined by the International Commission on Illumination (CIE)
- **CIEDE2000**  color difference defined by the International Commission on Illumination (CIE)
- **SVD**  Single Value Decomposition
- **LDA**  Linear Discriminant Analysis
- **GNB**  Gaussian Naive Bayes

**Appendix A. Light Source**

<table>
<thead>
<tr>
<th>Feature</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power</td>
<td>36 W</td>
</tr>
<tr>
<td>Color temperature</td>
<td>5800 K ± 300 K</td>
</tr>
<tr>
<td>Luminous flux</td>
<td>4200 lm</td>
</tr>
</tbody>
</table>
Appendix B. Quantitative Evaluation of 3DWF Set-Up

Appendix B.1. 3DWF Distance from Model to the Frontal Camera

Table A2. Evaluation of the distance from the model to the frontal camera.

<table>
<thead>
<tr>
<th>Distance (cm)</th>
<th>Cloud Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>7754</td>
</tr>
<tr>
<td>100</td>
<td>5871</td>
</tr>
<tr>
<td>120</td>
<td>4342</td>
</tr>
</tbody>
</table>

Appendix B.2. 3DWF Light Source Luminous Flux

Table A3. Evaluation of the influence of the light source luminous flux.

<table>
<thead>
<tr>
<th>Lm</th>
<th>Points of the Cloud</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Cloud Cam. 1</td>
</tr>
<tr>
<td>5000</td>
<td>5859</td>
</tr>
<tr>
<td>2625</td>
<td>6029</td>
</tr>
<tr>
<td>1435</td>
<td>6084</td>
</tr>
<tr>
<td>250</td>
<td>6621</td>
</tr>
</tbody>
</table>

Appendix B.3. 3DWF Light Source Orientation

Table A4. Evaluation of the orientation of the light sources.

<table>
<thead>
<tr>
<th>Focus</th>
<th>Angles</th>
<th>Light Cams</th>
<th>Cam 1</th>
<th>Cam 2</th>
<th>Cam 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Front</td>
<td>30°</td>
<td>40°</td>
<td>189</td>
<td>56</td>
<td>58</td>
</tr>
<tr>
<td>Side</td>
<td>40°</td>
<td>30°</td>
<td>138</td>
<td>51</td>
<td>51</td>
</tr>
<tr>
<td>Front</td>
<td>30°</td>
<td>60°</td>
<td>164</td>
<td>50</td>
<td>68</td>
</tr>
</tbody>
</table>

Appendix B.4. 3DWF Cameras Orientation

Table A5. Evaluation of the orientation of the cameras.

<table>
<thead>
<tr>
<th>Angle</th>
<th>Cam 1</th>
<th>Cam 2</th>
<th>Cam 3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\bar{I}$</td>
<td>$\bar{\sigma}_I$</td>
<td>$\bar{I}$</td>
</tr>
<tr>
<td>30°</td>
<td>129.35</td>
<td>62.69</td>
<td>94.22</td>
</tr>
<tr>
<td>40°</td>
<td>172.00</td>
<td>71.09</td>
<td>98.29</td>
</tr>
<tr>
<td>60°</td>
<td>171.50</td>
<td>71.90</td>
<td>108.674</td>
</tr>
</tbody>
</table>

References


© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).