FaceCept3D: Real Time 3D Face Tracking and Analysis

Sergey Tulyakov, Radu-Laurenţiu Vieriu, Enver Sangineto, Nicu Sebe
Department of Information Engineering and Computer Science
University of Trento, Italy
{sergey.tulyakov, enver.sangineto}@unitn.it, {vieriu, sebe}@disi.unitn.it

Abstract

We present an open source cross platform technology for 3D face tracking and analysis. It contains a full stack of components for complete face understanding: detection, head pose tracking, facial expression and action units recognition. Given a depth sensor, one can combine FaceCept3D modules to fulfill a specific application scenario. Key advantages of the technology include real time processing speed and ability to handle extreme head pose variations. Possible application areas of the technology range from human computer interaction to active aging platform, where precise and real-time analysis is required. The technology is available for scientific community.

Figure 1. Several examples of FaceCept3D applied to the RGB-Depth stream. The system requires a person specific template to operate. Typically, the template creation is performed in advance and takes only 3 seconds to complete.

1. Introduction

Over the past years, there has been an increasing interest in technologies aimed at supporting or enhancing people's lives (especially elderly class) in various environments, such as shopping malls, museums or at home [1, 2]. Understanding the affective state of these subjects offers important clues in decoding their state of mind, useful in monitoring tasks. In addition, many studies require estimates of the direction and level of attention for modeling different types of interactions. In such cases, the head pose estimation becomes a valuable proxy.

There is one important constraint all these scenarios share when looking for solving the above mentioned tasks: *non-invasiveness*, *i.e.* the solution must not hinder the naturalness of the subject's behavior. As a consequence, the vision sensors are typically placed out of the direct sight of the subject. FaceCept3D is motivated by challenges arising from these types of scenarios and is able to successfully address them in a unified, open source and cross-platform solution. Additionally, our system can be deployed in a much broader spectrum of applications (*e.g.* those cases for which the face is fully visible to the sensor), being able to maintain state-of-the-art performance, as shown in [34]. Several examples of correctly handled self-occlusions due to face rotations are given in the Figure 1. The code of the technology is available on GitHub[1].

2. Related work

Since our proposed system addresses multiple tasks in a unified solution, we will briefly review related work for each separate task, namely 3D head pose estimation, pose-invariant facial expression and facial action unit recognition.

2.1. Head pose estimation in 3D

There is an abundance of work ([21]) exploiting 2D information for estimating head pose (HPE). However, 2D acquisition devices are sensitive to illumination changes, making it difficult to find simple enough features to meet real-time constraints. Naturally, depth images are not influenced by illumination and, as such, become an attractive alternative for HPE. As a consequence, many research groups have investigated the use of 3D data,

[1] https://github.com/sergeytulyakov/FaceCept3D

especially since sensors such as Kinect have become affordable.

One way of addressing HPE in 3D is to treat the task as a mesh registration problem. Weise et al. [36] present a method for transferring facial expressions from a user to an animated avatar. They first create a person-specific model for a performer by manually marking correspondences between the model and the user. This operation is done automatically in [15], eliminating the requirement to do offline tuning of the system to a particular human performer. However, methods like [36, 29, 17, 7] still rely on landmarks detection, such as eyes, nose, mouth and other facial points. In the context of severe head orientations, accurate correspondence estimation is no longer possible (due to self occlusions) and the accuracy of the system decreases. We too build a personalized template for a given user, but in contrast to previous work, our template creation stage is offline, fully automatic and estimating the head pose needs no facial landmarks available.

In [13], an approach for head pose estimation based on random regression forests is described. The results obtained on Biwi Kinect Database are promising (the percentage of correctly identified test examples within the threshold of 10° degrees is 90.4%), however, the case of extreme head orientations is not covered in their study. In [6] the authors propose a novel shape signature to help identifying nose position in range images. Using parallel computing, they evaluate many pose hypotheses reaching a hit rate of 97.8% corresponding to an error threshold of 15° at $55.8\,fps$. In the same context, in [22] a particle swarm optimization search generates remarkably small uncertainty when predicting head pose (around 2° standard deviation for all angles), but, similar to [6], they also resort to massive parallel resources coming from GPU.

In order to handle large pose variations and process non-frontal facial views, we fuse two independent components: a detection module and a tracking one. In addition, by keeping the pipeline simple, no dedicated hardware is needed to speed up processing, reaching real-time performance on standard CPUs.

2.2. Pose-invariant facial expression recognition

In the past decade, much work has been done on static facial expression recognition dealing with non frontal poses by exploiting 3D data. Such methods are split in [25] into four main categories: *distance based*, *patch based*, *morphable models* and *2D representations*.

Distance based methods extract the (3D) landmark positions of the input face and use inter-landmark distances to classify facial expressions [28, 16]. Obvious limitations arise from the difficulty of localizing facial landmarks in cases of severe self-occlusions. Patch based approaches extract local features from either every point of a 3D mesh or around specific landmarks [25]. For instance, in [18] facial landmarks on the 3D surface of a face specify the positions in which patches are described by means of level curves. Probe and gallery expression samples are compared computing the geodesic distance between such curves. Note that in our approach we extract patches from the 2D projection of the 3D face point cloud representation. In [20] a morphable model is fitted to the face point cloud by matching a set of landmarks, which need to be localized both on the prototypical model and on the analyzed face.

The 2D approaches [25] are the category most similar to our method and are based on mapping the 3D data onto 2D representations. Once the mapping is computed, different features can be extracted from the 2D representation. For instance, in [24] depth maps and Azimuthal Projection Distance Images are filtered with different methods, such as Gabor filters, LBP features, etc., with the goal of action unit detection. In [4] a depth map of the 3D facial meshes is computed and SIFT features are extracted in this map around specific landmarks. In our approach we do not need to accurately localize landmarks on our 2D representation and a rough estimation of the head pose together with the position of the eyes in the depth map is sufficient to compute our cylindrical projection surface. Moreover, our Random Forest based joint selection of features and channels makes it possible to adaptively choose among a huge number of possible features.

Closest to our system in systematically addressing severe head orientations is the work of [23] in which the authors use a discrete set of 34 poses (spanning angles between -45° and 45° on the *yaw* direction and between -30° and 30° for the *tilt*) along with the frontal pose in order to learn a mapping function between facial landmark positions of a given non-frontal face and the frontal correspondence. At testing time, the head pose is estimated and the closest training poses are used to project the landmarks onto the frontal pose. Finally, a multi-class SVM is applied to the frontally-normalized landmark positions to categorize the facial expressions. One drawback of this work is the necessity to accurately estimate a large number of facial landmarks. Many systems estimating such landmarks fail when large head pose variations come into play and, as such, alter drastically the performance of subsequent processing stages.

2.3. Facial Action Unit recognition

Along with facial expression recognition, action unit analysis has been in the center of attention of many research groups. Action units (AU) are anatomical descriptors which correspond to various facial muscle contractions. They can occur alone or in hundreds of combinations which account for all possible facial expressions. In other words, facial expressions in general (and Eckman's 6 prototypical ones in

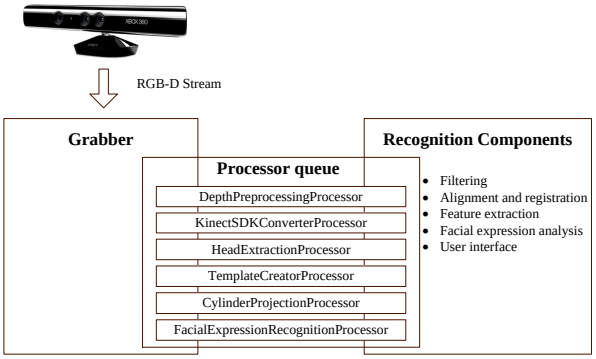

Figure 2. A pipeline for tracking the head pose and recognizing facial expressions. Processors are executed by the grabber one after another.

particular) are manifestations of a small subset of AUs. The most commonly used taxonomy for action units is described by FACS [12] and typically research focuses on a small subset only.

There is a large body of research on AU recognition from both 2D and 3D data (see recent surveys for reference [39, 11, 26, 19]). Approaches are generally split into static and dynamic ones. In the first category, AU recognition is posed as a binary (*1-vs-all*) or multi-class classification problem, using different features, typically extracted around salient facial landmarks (such as SIFT [9, 42, 38] or LBP-based [3, 24]) and different classifiers (SVM, AdaBoost). In dynamic modeling, frames are grouped into sequences and temporal models (for instance HMMs, CRFs [8, 27, 33]) are used to dissociate between AU components, such as *onset, apex* and *offset*.

More recently [35, 42], efforts have been channeled into modeling dependencies between combinations of AUs, showing improved performance *w.r.t.* the simpler models. In [42] a joint patch and multi-label learning framework for AU recognition is being proposed, in which dependencies between certain pairs of AUs are modeled using a matrix that encodes *positive correlations* and *negative competitions* computed from an ensemble of datasets. It shows superior results over both methods that focus only on learning patch importance [30, 43] as well as those adopting the multi-label strategy [40].

As in the case of face analysis in general, action unit recognition is now addressing spontaneous scenarios (i.e. cases in which data are generated following carefully planned elicitation protocols, including subjects that have not been particularly trained or prepared for the task), moving one step closer to real life situations ([41]). Recent papers and challenges [32] are now publishing results along this line.

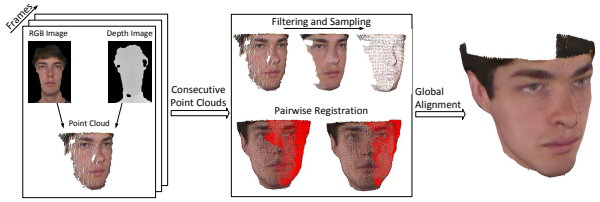

Figure 3. Person-specific template creation pipeline. Prior to creating the point cloud, we filter out noise by convolving the depth image with a Gaussian kernel. Voxel-grid algorithm on the smooth cloud is used to obtain a cloud with fewer points. Pairwise registration is performed on the consecutive clouds.

3. Modular architecture

FaceCept3D is a set of independent modules. All modules are split into three major parts:

- **Recognition modules** include filtering, registration, feature extraction, machine learning methods and other components.
- **Pipeline modules**, that encapsulate underlined platform and sensor-specific technical details.
- **User interfaces modules**, that enable viewing, annotating and displaying the results.

Figure 2 shows a typical pipeline for an automatic head pose tracking and facial expression recognition. A sensor dependent grabber module executes a queue of processors that perform necessary actions using the recognition components.

4. Head pose tracking

In order to track a face, FaceCept3D builds offline a person-specific 3D head template for a person in front of the sensor. When the template is ready a modified version of the Iterative Closest Point (ICP) [5] method is used to register it with a scene and obtain the head pose (more details in [31]). The process of person-specific template creation is outlined in Figure 3 and takes around 3 seconds on a embedded Intel processor.

Our modified version of the ICP algorithm uses history-based points weighting as described in [31] to guide the optimization procedure of ICP to a promising descend direction and reach local minima faster. Table 1 shows that our version of ICP converges almost 4 times faster. Several examples of recognized head poses are given in the Figure 6. Note the difficult viewing and head orientation correctly handled by the system.

To evaluate head pose tracking accuracy we use the Dali3DHP RGB-D dataset [31]. This database contains two sessions of range and RGB images of 33 subjects.

Table 1. Comparison between history-based weighted ICP and generic ICP in computational time

	# Iterations	Fps
Generic ICP	14.64	10.05
History-based weighted ICP	**3.16**	**38.87**

During the first session a subject is asked to perform a left-to-right head movement. This session is used to create a person specific head template. During the second session, a subject performs a set of head movements. To ensure uniform distribution over all possible head pose ranges, all the subjects follow a special head movement pattern on the wall. Ground truth is recorded using a Shimmer sensor [2]. Table 2 shows the results. Note that weighted template tracker provides a slightly better results than the full template tracker, while maintaining 4 times faster speed.

Table 2. Head pose estimation result obtained on Dali3DHP Dataset. Mean average error and the standard deviation (in brackets) are reported.

	yaw	tilt
Full Template	4.06 (5.89)	8.21 (11.45)
Weighted Template	3.93 (5.23)	8.21 (11.31)

4.1. Head pose invariant face representation

FaceCept3D head pose tracker returns head pose orientation in real-time. Since subjects are not constrained in head movements, many parts of the face could be self-occluded. Therefore a head pose invariant representation is required. We build such representation by constructing a cylinder around the face and projecting the face onto the cylinder. Figure 4 shows this cylindrical sampling pipeline.

Several examples of head pose invariant face representation are given in Figure 4 bottom row. Note how the head pose problem is transformed into a missing information problem. Nearest neighbor interpolation is the most computationally expensive step in this pipeline. In order to run it in real-time FaceCept3D has an efficient way to compute it.

5. Facial Expression and Action Unit Recognition

Once computed, the head pose invariant face representation is subject to a dense sampling procedure with overlapping patches of fixed size (see Figure 5). For each patch position, we train a separate classifier, followed by a late fusion stage for the final estimate. In the case of

[2]http://shimmersensing.com/

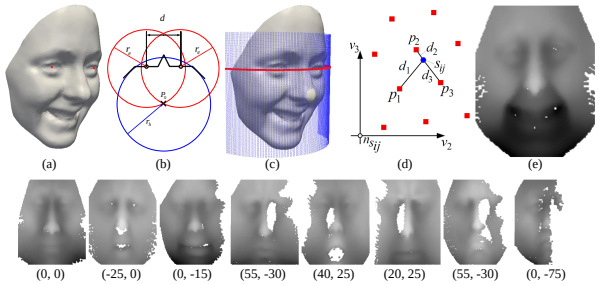

Figure 4. Top row (left to right): an example of face scan with two eyes detected. The cylindrical head model (CHM) parameters. Our CHM with 150×120 sampling points imposed on the face scan. Sampling point values computed based on the 3 nearest neighbors. An example of pose-invariant face representation. Bottom row: examples of sampled faces under varying head poses and facial expressions. The head rotation $(tilt, yaw)$ is given in the brackets.

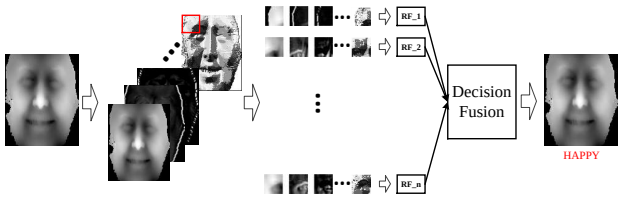

Figure 5. From head pose invariant face representation to the expression label (left to right): initial 2D face representation, channel computation, dense sampling with overlapping patches, random forest classification, decision fusion and labeled sample.

action unit (AU) recognition, we employ a 1-vs-all strategy for every patch. The dense sampling approach comes along with two important benefits: (i) it offers an elegant way to cope with missing information, as the *empty* patches are simply discarded at decision making stage and (ii) it is naturally suited for modeling patch importance, as different patch votes can be weighted differently (especially in the case of AU recognition).

From each face image encoding depth information (*i.e* each pixel value reflects the distance between the object and the sensor), we first compute channel representations [10], then we split the channels into overlapping patches, from which generalized Haar features are extracted. Random Forests are then used to perform patch level predictions, which in turn are aggregated for the final estimate [34].

Figure 7 shows the recognition rate distribution over the yaw/tilt space on BU-3DFE dataset [37]. The angle ranges are divided into blocks of equal size $15^\circ \times 15^\circ$ and performance is computed on samples belonging to each block. The gray area corresponds to a reduced set of angles, commonly used in previous work (*e.g.* [23]). While maintaining state-of-the-art performance on the reduced set, FaceCept3D is able to extend its operating point to severe head rotation angles with only a reasonable loss in

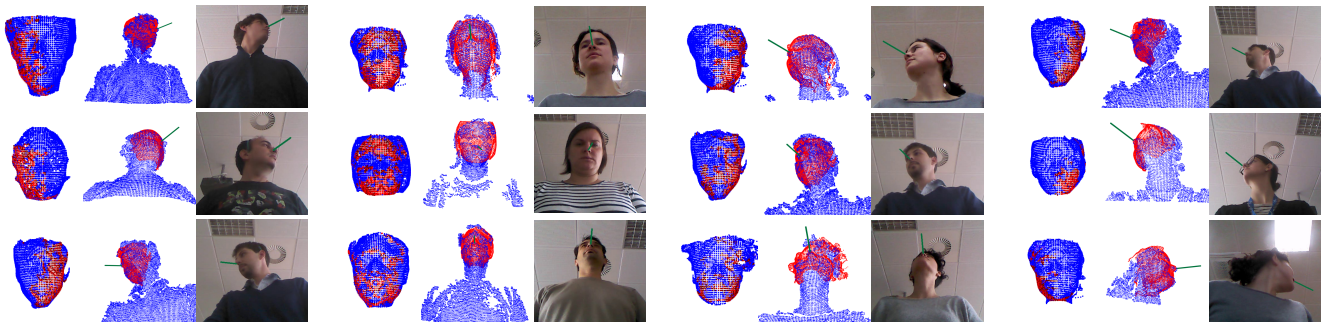

Figure 6. For every subject three images are given. The left one represents a template with the most important points marked in red. The image in the middle shows the template fitted to the point cloud. The right image shows the view from the walker. Note that for some subjects the face is almost completely hidden.

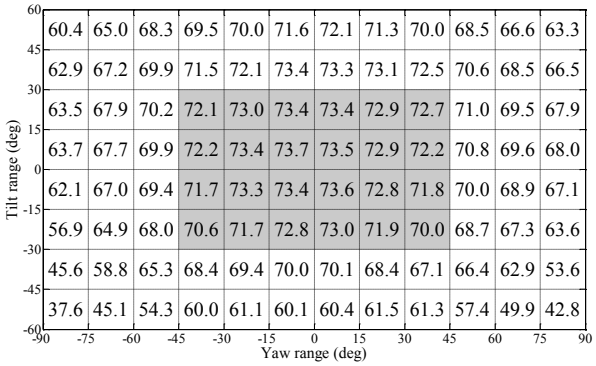

Figure 7. Recognition rate distribution over the yaw/tilt space. The gray area shows the reduced head-pose range reported in [23].

recognition accuracy.

Table 3. Action Unit recognition results obtained on BP4D

AU Index	F1 Norm	Acc Norm
1	0.46	0.60
2	0.12	0.50
4	0.36	0.56
6	0.80	0.79
7	0.73	0.70
10	0.79	0.77
12	0.82	0.81
14	0.68	0.66
15	0.33	0.56
17	0.58	0.63
23	0.43	0.60
Avg	**0.56**	**0.65**

Finally, in Table 3, we show preliminary results on AU recognition on BP4D dataset [41], following a leave-one-subject-out protocol. As a performance measure, we report the normalized F1 score with a skew factor [14], computed as $F1Norm = \frac{2sPR}{2sR+P}$, where R and P are the Recall and Precision, respectively, and s is the ratio between the number of negative samples and the number of positive ones included in the test set. In a similar manner we compute the skew-normalized accuracy, as $AccNorm = \frac{TP+TN/s}{TP+TN/s+FP/s+FN}$.

6. Conclusions

In this paper we introduce FaceCept3D, an open source cross platform system for 3D face analysis. FaceCept3D is able to accurately infer head pose, perform face frontalization and estimate facial expressions in real-time. Our system is designed to cope with a wide range of head pose variations, typically seen in applications for which non-invasiveness is a particularly important requirement.

References

[1] `http://cordis.europa.eu/project/rcn/101220_en.html`.

[2] `http://cordis.europa.eu/project/rcn/194087_en.html`.

[3] N. Bayramoglu, G. Zhao, and M. Pietikainen. Cs-3dlbp and geometry based person independent 3d facial action unit detection. In *ICB*, pages 1–6, 2013.

[4] S. Berretti, B. B. Amor, M. Daoudi, and A. D. Bimbo. 3d facial expression recognition using sift descriptors of automatically detected keypoints. *The Visual Computer*, 27(11):1021–1036, 2011.

[5] P. Besl and N. D. McKay. A method for registration of 3-D shapes. *PAMI*, 14(2):239–256, 1992.

[6] M. D. Breitenstein, D. Kuettel, T. Weise, L. Van Gool, and H. Pfister. Real-time face pose estimation from single range images. In *CVPR*, pages 1–8, 2008.

[7] K. I. Chang, W. Bowyer, and P. J. Flynn. Multiple nose region matching for 3d face recognition under varying facial expression. *PAMI*, 28(10):1695–1700, 2006.

[8] K.-Y. Chang, T.-L. Liu, and S.-H. Lai. Learning partially-observed hidden conditional random fields for facial expression recognition. In *CVPR*, pages 533–540, 2009.

[9] W. S. Chu, F. D. L. Torre, and J. F. Cohn. Selective transfer machine for personalized facial action unit detection. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013.

[10] P. Dollár, Z. Tu, P. Perona, and S. Belongie. Integral channel features. In *BMVC*, 2009.

[11] S. Du, Y. Tao, and A. M. Martinez. Compound facial expressions of emotion. *National Academy of Sciences*, 111(15):E1454–E1462, 2014.

[12] P. Ekman and W. V. Friesen. Facial action coding system. 1977.

[13] G. Fanelli, M. Dantone, J. Gall, A. Fossati, and L. Gool. Random Forests for Real Time 3D Face Analysis. *IJCV*, 101(3):437–458, 2012.

[14] L. Jeni, J. F. Cohn, F. De La Torre, et al. Facing imbalanced data–recommendations for the use of performance metrics. In *ACII*, pages 245–251, 2013.

[15] H. Li, J. Yu, Y. Ye, and C. Bregler. Realtime facial animation with on-the-fly correctives. In *SIGGRAPH*, volume 32, page 1, 2013.

[16] X. Li, Q. Ruan, and Y. Ming. 3D facial expression recognition based on basic geometric features. In *ICSP*, 2010.

[17] X. Lu and A. K. Jain. Automatic feature extraction for multiview 3d face recognition. In *AFGR*, pages 585–590, 2006.

[18] A. Maalej, B. B. Amor, M. Daoudi, A. Srivastava, and S. Berretti. Shape analysis of local facial patches for 3d facial expression recognition. *Pattern Recognition*, 44(8):1581–1589, 2011.

[19] A. Martinez and S. Du. A model of the perception of facial expressions of emotion by humans: Research overview and perspectives. *JMLR*, 13(1):1589–1608, 2012.

[20] I. Mpiperis, S. Malassiotis, and M. G. Strintzis. Bilinear models for 3-d face and facial expression recognition. *Information Forensics and Security*, 3(3):498–511, 2008.

[21] E. Murphy-Chutorian and M. M. Trivedi. Head Pose Estimation in Computer Vision: A Survey. *PAMI*, 31(4):607–626, 2009.

[22] P. Padeleris, X. Zabulis, and A. A. Argyros. Head pose estimation on depth data based on particle swarm optimization. In *CVPRW*, pages 42–49, 2012.

[23] O. Rudovic, M. Pantic, and I. Patras. Coupled gaussian processes for pose-invariant facial expression recognition. *PAMI*, 35(6):1357–1369, 2013.

[24] G. Sandbach, S. Zafeiriou, and M. Pantic. Binary pattern analysis for 3d facial action unit detection. In *BMVC*, 2012.

[25] G. Sandbach, S. Zafeiriou, M. Pantic, and L. Yin. Static and dynamic 3d facial expression recognition: A comprehensive survey. *IVC*, 30(10):683–697, 2012.

[26] G. Sandbach, S. Zafeiriou, M. Pantic, and L. Yin. Static and dynamic 3d facial expression recognition: A comprehensive survey. *IVC*, 30(10):683–697, 2012.

[27] L. Shang and K.-P. Chan. Nonparametric discriminant hmm and application to facial expression recognition. In *CVPR*, pages 2090–2096, 2009.

[28] H. Soyel and H. Demirel. Facial expression recognition using 3d facial feature distances. In *ICIAR*, 2007.

[29] Y. Sun and L. Yin. Automatic pose estimation of 3d facial models. In *ICPR*, pages 1–4, 2008.

[30] S. Taheri, Q. Qiu, and R. Chellappa. Structure-preserving sparse decomposition for facial expression analysis. *TIP*, 23(8):3590–3603, 2014.

[31] S. Tulyakov, R. L. Vieriu, S. Semeniuta, and N. Sebe. Robust Real-Time Extreme Head Pose Estimation. In *ICPR*, 2014.

[32] M. Valstar, J. Girard, T. Almaev, G. McKeown, M. Mehu, L. Yin, M. Pantic, and J. Cohn. Fera 2015-second facial expression recognition and analysis challenge. *IEEE ICFG*, 2015.

[33] M. F. Valstar and M. Pantic. Fully automatic recognition of the temporal phases of facial actions. *Systems, Man, and Cybernetics*, 42(1):28–43, 2012.

[34] R.-L. Vieriu, S. Tulyakov, S. Semeniuta, E. Sangineto, and N. Sebe. Facial expression recognition under a wide range of head poses. In *FG*, 2015.

[35] Z. Wang, Y. Li, S. Wang, and Q. Ji. Capturing global semantic relationships for facial action unit recognition. In *ICCV*, pages 3304–3311, 2013.

[36] T. Weise, S. Bouaziz, H. Li, and M. Pauly. Realtime performance-based facial animation. In *SIGGRAPH*, 2011.

[37] L. Yin, X. Wei, Y. Sun, J. Wang, and M. J. Rosato. A 3D facial expression database for facial behavior research. In *FG*, 2006.

[38] A. Yuce, H. Gao, and J.-P. Thiran. Discriminant multi-label manifold embedding for facial action unit detection. In *FG, FERA 2015 Challenge*, 2015.

[39] Z. Zeng, M. Pantic, G. I. Roisman, and T. S. Huang. A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *PAMI*, 31(1):39–58, 2009.

[40] X. Zhang, M. H. Mahoor, S. M. Mavadati, and J. F. Cohn. A l p-norm mtmkl framework for simultaneous detection of multiple facial action units. In *WACV*, pages 1104–1111, 2014.

[41] X. Zhang, L. Yin, J. F. Cohn, S. Canavan, M. Reale, A. Horowitz, and P. Liu. A high-resolution spontaneous 3d dynamic facial expression database. In *FG*, pages 1–6, 2013.

[42] K. Zhao, W.-S. Chu, F. De la Torre, J. F. Cohn, and H. Zhang. Joint patch and multi-label learning for facial action unit detection. In *CVPR*, pages 2207–2216, 2015.

[43] L. Zhong, Q. Liu, P. Yang, B. Liu, J. Huang, and D. N. Metaxas. Learning active facial patches for expression analysis. In *CVPR*, pages 2562–2569, 2012.