Obtaining a 3D Model from a Facial Recognition in 2D

G. Peláez, F. García, A. de la Escalera, and J.M. Armingol

Systems Engineering and Automation Department,
Intelligent Systems Laboatory, University Carlos III of Madrid, Leganes
{gpelaez,fegarcia,escalera,armingol}@ing.uc3m.es
http://www.uc3m.es/islab

Abstract. This paper shows the current status of an implementation with a composed device of depth and color camera. From the color image, a set of points associated with the face is obtained; later the main features of a human face are identified. The 3D model is constructed based on a previous 2D analysis using the haar-like features for detecting the human face. This application will be a part of a more complex system designed to assist the driver by monitoring both inside and outside the vehicle, i.e. intelligent systems of transportation.

Keywords: Facial recognition, 3D perception, driving assistance, intelligent vehicles.

1 Introduction

Facial recognition is a popular method used in these days for many different applications. Normally, a red-green-blue (RGB) image and a specific library is all what is needed to perform this method. Unfortunately, when dealing with object recognition and pattern matching, some adversities must be overcome in order to obtain a robust detection-recognition application. One of the main problems is the dependence on illumination conditions that will affect the perception of colors and shape detection. It is described in this paper, the usage of the XBOX 360 Kinect from Microsoft to show a first approach on how to obtain additional information for facial recognition besides the color image. This is possible thanks to the depth perception feature in the Kinect. From the depth perception, a 3D model of the face will be obtained.

2 State of the Art

Although there are some facial recognition experiments and the tools are easily accessible, not so many of this experiments take the depth information into account, this because of hardware restrictions or because it was not but until now, with the release of the Kinect, that an affordable hardware setup was available to obtain color images and 3D information at the same time.

R. Moreno-Díaz et al. (Eds.): EUROCAST 2013, Part I, LNCS 8112, pp. 33–38, 2013.

2.1 Facial Recognition

Today the study of the facial features is a branch that has a lot of work behind and many approaches and experiments have been published in different scientific journals and magazines. Although the final goal may differ, many of the basic points are shared, also the difficulties involved that can vary from one experiment to another for example, a rapid 3D face modeling is achieved in [1] using two images. The use of the second image is justified on the fact that more accurate depth information can be obtained from it. By merging the data obtained from both images, a 3D shape is obtained. To finish, the intensity of the frontal image is mapped on the reconstructed 3D shape for a realistic solution. Another project proposes an algorithm for facial feature extraction and recognition based on a Curvelet transform and singular value decomposition [2]. It does a Curvelet transformation first, extracts the Curvelet energy of low frequency and high frequency later. It follows by doing a singular value compression and fusions the facial features to this component. It ends using the nearest neighbor classifier in the ORL person face database to confirm the validity of the algorithm. Finally, the detection of drowsiness in real time is achieved in [3] where a system based on visual information and artificial intelligence is proposed to assist the driver. The algorithm also locates and tracks the face and eyes to compute a drowsiness index. The main goal is to help to reduce a possible car accident due to drowsiness or distraction from the driver.

2.2 Kinect Projects

Some projects have been developed with the Kinect after the release of the first drivers available through the result of a competition partially sponsored by Adafruit Industries [4] by the end of 2010. A month later PrimeSense[5], who designed the depth sensing in the Kinect, released their own drivers. Now, a large set of experiments have been developed in different branches. Many others are available in web pages such as Hackaday[6]. Projects with Kinect involve a wide variety of applications: As demonstrated in [7], the tracking of a hand is useful for applications oriented to virtual reality. Some others like [8] provide a solution that uses gestures as a way to interact with virtual objects in an augmented reality application. In [9] the detection of a human presence is achieved with a Kinect by using the depth information and 2D information associated to the head's contour. The Kinect can also be used as a complementary sensor in a more complex system such as in [10] where a mobile robot uses, as part of its localization system in an indoor environment, a Kinect for the location of landmarks to correct the robot's position.

2.3 Time of Fly and Data Fusion Cameras

The Kinect is not the only time of fly sensor available for research, there are others and different applications have been developed such as a[11] where a touch less user interface is developed for certain medical applications that require a physical contact but should be touch less due to the sterility requirement in the operation's room. Another application is found in [12] that explains the development of a simulation of camera-like time-of-flight sensors. In [13], the time of fly and data fusion features are used in order to improve the road safety by detecting pedestrians.

3 System Description

The system is composed of one XBOX 360 Kinect from Microsoft. The Kinect is a sensor bar composed by a set of different sensors that can be separated into 3 categories: video, audio and motion. For the video part there is a time-of-fly camera, an RGB camera and one IR receiver. For the audio part, there are 4 microphones configured as an array across the Kinect sensor bar. And for the motion part there is one servo motor and a 3 axis accelerometer. Different open source libraries were used with this hardware. A deeper description follows.

4 System Functionality

Overall, the application will obtain a single structure through the Point Cloud Library (PCL). It contains information about the video and depth perceived. From this structure (1), the RGB information will be extracted to build a 640x480 picture. The cloud, obtained from the sensor (Fig. 1), is a point structure representing Euclidean xyz coordinates according to a reference system where the center of the cloud corresponds to “0” in each of the 3 axis, and the RGB color (1). P is the point cloud, x,y and z define the Euclidian coordinates and c is the color channel. Each point corresponds to one projection of the time of fly camera. For this project, a cloud with points PointXYZRGB is used i.e. a cloud with the Euclidean coordinates and color information for each point.

$$P = \sum_{x=0}^{w} \sum_{y=0}^{h} \sum_{z=0}^{d} \sum_{c=1}^{3} p_{xyzc} \tag{1}$$

Once the cloud is obtained, the color image is built from the cloud by unpacking the RGB values from the cloud to a matrix with the same dimensions as the cloud and 3 channels, one for each color as seen in (2) where M is the resulting matrix; p is one point in M; w and h are the height and width of the image respectively; c is the color channel. This matrix is then converted to an image where the vision algorithms can be applied. A direct association between a detected object in the color image and its corresponding set of points in the cloud can be done with no additional conversions.

$$M = \sum_{x=0}^{w} \sum_{y=0}^{h} \sum_{c=1}^{3} p_{xyc} \tag{2}$$

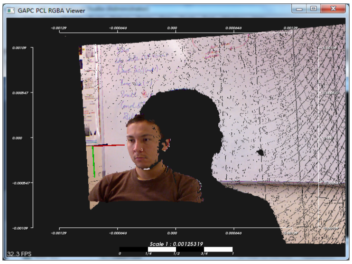

Fig. 1. XYZRGB cloud with the coordinates system

The image is now constructed and the next step is to search for the facial features in it. To address this problem, a series of search procedures will be done with different ROIs. This means that the overall image will be subject to a search of a larger object and once detected, another search, more specific, will be done but only in the area corresponding to the previous large object detected.

Notice that if this embedded search was not performed and instead a direct search for eyes and mouth was done, it would take more as the searching algorithm will take more time searching for a small object in a large image compared to searching for a larger object in the same image. Therefore, the application will not go straight for the facial features but instead will go for a constant reduction of the searching space followed by more specific searches. Inside the image, the first object to be searched is the upper body, then the face and finally the eyes and mouth. A haar-like features method was used to perform the detection. This method allows the search of a part of the human body in the whole picture. The algorithm starts with a window of determined size that is moved over the input image, and for each subsection of the image the object specified in the configuration file is searched. If the whole image is analyzed and no object was found, the area of the window is increased by a percentage specified in the parameters of configuration and the search is repeated. So, to find an object of an unknown size in the image the scan procedure should be done several times at different scales. Only the upper body, face, eyes and mouth were used for this project. The ROI (3) of the resulting object has different properties that will be used later such as width (w'), height (h'), and the coordinates of the upper left corner. This ROI will be a sub-image and therefore will also have the 3 color channels (c') with it.

$$ROI = \sum_{x=0}^{w\prime} \sum_{y=0}^{h\prime} \sum_{c=1}^{3} p_{xyc} \tag{3}$$

After the facial features are obtained, a new cloud is constructed (4) only by those points that belong to the face. This is, all the p_{xyzc} that are contained inside (3). Notice that the depth parameter z is not filtered. P' is the point cloud associated to the face. As a result, a point cloud containing information about the depth and color of the face is available. This model can be stored in a particular file format for a future analysis. The resulting model has the same components as the original cloud from where it was extracted.

$$P' = \sum_{x=0}^{w\prime} \sum_{y=0}^{h\prime} \sum_{z=0}^{d} \sum_{c=1}^{3} p_{xyzc} \tag{4}$$

Although it's possible to obtain more than one face and its respective point cloud, only one is obtained at this stage of the project due to its orientation which aims at supervising the driver's gestures and gaze direction.

5 Results

The system was tested with different lighting conditions and positions of the face. In all cases, where the face was present and detected in the RGB image, a PCL from the face was obtained. The speed of the algorithm will depend on the parameters of configuration of the face detection library.

Although this device was designed for analyzing objects the size of a human body, it delivered good results as the facial 3D model has some distinguishable features such as the nose and the ocular cavities as seen in Fig.2. Another advantage obtained from this system is the possibility of the reduction of false positives when detecting faces inside the RGB image because a simple mask of the 2D image with the depth image would deliver a much reduced search space for the face and therefore eliminating false candidates that have geometrical and chromatic similitudes with a face.

Overall, it took between 0.2 and 0.4 seconds to identify and obtain the colored cloud of the face and the correct detection of the other facial features (eyes and mouth basically) with a core i5 at 2Ghz. The resulting cloud has the same information as a 2D image would have but it also includes the depth perceived by the sensor and therefore offering a new approach for detecting and monitoring facial gestures. Applications that work as a driver's assistance where the face is constantly analyzed could benefit from the additional information of the depth perceived.

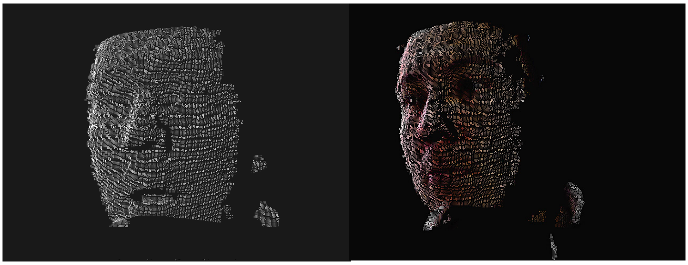

Fig. 2. Point Clouds of the face, XYZ and XYZRGB

6 Conclusion and Future Development

There are different ideas to continue this experiment, as there are many branches where more development is desirable according to the goal of the project. The main goal is to create a 3D model of the face, similar to the Active Appearance Models (AAM)in 2D. This will allow the introduction of the Kinect inside a vehicle for the supervision of the driver's gestures such as drowsiness, distraction and a first analysis of the environment if the image sensors are complemented with the array of microphones to analyze the amount of noise surrounding the driver.

Acknowledgments. This work was supported by the Spanish Government through the Cicyt projects (GRANT TRA2010-20225-C03-01) and (GRANT TRA 2011-29454-C03-02). CAM through SEGVAUTO-II (S2009/DPI-1509).

References

1. Heo, J., Savvides, M.: Rapid 3D face modeling using a frontal face and a profile face for accurate 2D pose synthesis. In: International Conference on Automatic Face & Gesture Recognition and Workshops, pp. 632–638 (March 2011)
2. He, J., Zhang, X.: Facial feature extraction and recognition based on Curvelet transform and SVD. In: International Conference of Apperceiving Computing and Intelligence Analysis, ICACIA 2009, pp. 104–107 (2009)

3. Flores, M.J., Armingol, J.M.: A Escalera: Real-time drowsiness detection system for an intelligent vehicle. In: IEEE Intelligent Vehicles Symposium, pp. 637–642 (June 2008)
4. Industries, Adafruit. Adafruit Industries (March 2012), `http://www.adafruit.com`
5. Industries, Prime Sense (March 2012) Prime Sense Industries, `http://www.primesense.com`
6. Hack a day community. Hack a Day (March 2012), `http://www.hackaday.com`
7. Frati, V., Prattichizzo, D.: Using Kinect for hand tracking and rendering in wearable haptics. In: World Haptics Conference (WHC), pp. 317–321 (June 2011)
8. Santos, E.S., Lamounier, E.A., Cardoso, A.: Interaction in Augmented Reality Environments Using Kinect. In: 2011 XIII Symposium on Virtual Reality (SVR), pp. 112–121 (May 2011)
9. Xia, L., Chen, C.-C., Aggarwal, J.K.: Human detection using depth information by Kinect. In: Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 15–22 (June 2011)
10. Ganganath, N., Leung, H.: Mobile robot localization using odometry and kinect sensor. In: International Conference on Emerging Signal Processing Applications (ESPA), pp. 91–97 (January 2012)
11. Soutschek, S., Penne, J., Hornegger, J., Kornhuber, J.: 3-D gesture-based scene navigation in medical imaging applications using Time-of-Flight cameras. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, pp. 16–23 (June 2008)
12. Keller, M., Orthmann, J., Kolb, A., Peters, V.: A Simulation Framework for Time-Of-Flight Sensors. In: International Symposium on Signals, Circuits and Systems, vol. 1, pp. 1–4 (2007)
13. Garcia, F., de la Escalera, A., Armingol, J.M., Herrero, J.G., Llinas, J.: Fusion based safety application for pedestrian detection with danger estimation. In: Proceedings of the 14th International Conference on Information Fusion, pp. 1–8 (July 2011)