The 13th International Scientific Conference
eLearning and Software for Education
Bucharest, April 27-28, 2017
10.12753/2066-026X-17-000

Face Recognition for Education in the Cloud

George-Alex Stelea, Cristinel Gavrilă, Sorin Zamfir, Radu Curpen
Department of Electronics and Computers, Transilvania University of Brasov, Romania,
george.stelea@unitbv.ro ,cristinel.gavrila@unitbv.ro,zamfir.sorin@unitbv.ro, radu.curpen@unitbv.ro

Abstract*: Over the last decade, face recognition has become a popular area of research and with the cloud computing paradigm gaining more and more ground, thus ensuring the resources needed for real-time image/video analysis, new and more powerful face detection algorithms have been developed. Face recognition has a wide spectrum of applicability such as: access control, identification systems, surveillance or monitoring group emotions and combined with real-time and cloud-computing paradigms has led to the development of various successful systems used by important entities (SmartGate – Australian Customs Services, Next Generation Identification –FBI, Visidon Applock – Android). The main advantage of the face recognition is that it is a non-contact process thus, it is able to offer a wide rage of information about a subject or group of subjects without requiring explicit actions from the target except the presence. The paper presents the face recognition for educational purpose, with the challenges and opportunities of cloud computing. The perspective of usability is more on intelligent testing based on real-time communications, with thin clients and face display embedded in usual browsers but with strong data analytics in the cloud, for authentication or classification.*

Keywords: *ODL, Video-conferencing; Open Face; Machine Learning; WebRTC; Tensorflow; Microsoft Azure Cognitive Services.*

I. INTRODUCTION

"Education as a Service" is a great challenge for the Cloud. Human presence cannot be always replaced by avatars based on virtual emotional agents for intelligent tutoring systems. There are many functionalities of ODL (Online and Distance Learning) that require face recognition for real participants in educational video-conferencing. This paper presents a range of contributions to trainers-trainees recognition based on newest technologies for real-time cloud communications (WebRTC) and data analytics [1]. The concept of face recognition is frequent in security systems and comparisons are done with other biometric data provided by fingerprint recognition or eye iris recognition [2]. Recently, face recognition has became popular as a commercial identification and marketing instrument hence it is able to play a key role in eLearning systems.

II. EDUCATIONAL USE-CASES

Faces are recorded from the very beginning of educational processes. An ID card, for example, is issued just from the subscription of personnel, in any roles of a school – with or without a LMS (Learning Management System). An important use-case is the authentication – for any official phase of intelligent testing for instance. VUE – Virtual University Environments should benefit extensively of face recognition mainly for certification exams but also for WebRTC-based recording of any personal contribution in a multi-party video-debate (open class).

III. TECHNOLOGIES AND DEMONSTRATOR

The solutions developed by the authors are cloud-based, with "thin clients" (tablets or smartphones) and without localized computational dependencies – for instance, simple WebRTC camera streaming[3] towards display in usual browsers leaves all background processing to cloud data analytics based on Machine Learning[4] / Deep Learning[5]. Recognition is implemented with Tensorflow[6] / OpenFace[7]deep neural network methods / Microsoft Azure Cognitive Services [8], using Python, Torch, OpenCV and Face API.

3.1 OpenFace

OpenFace is an open-source face recognition solution using deep neural networks implemented in Torch and Python developed by Brandon Amos, Bartosz Ludwiczuk, and Mahadev Satyanarayanan at the Carnegie Mellon University, Pittsburgh, Pennsylvania U.S.A.

Open Face workflow overview:

1. Facial detection using a pre-trained model (OpenCV);
2. Face transformation for neural networks;
3. Face representation on a 128-dimensional hypersphere using deep neural networks.

Open face can be downloaded from GitHub[7] and can be installed manually or via Docker using the following command:

```
docker build -t openface .
docker run -p 9000:9000 -p 8000:8000 -t -iopenface/bin/bash
cd /root/openface
./run-tests.sh
./demos/compare.py images/examples/{lennon*,clapton*}
./demos/classifier.py infer models/openface/celeb-
classifier.nn4.small2.v1.pkl ./images/examples/carell.jpg
./demos/web/start-servers.sh
```

To run properly, OpenFace needs some prerequisites such as Python and OpenCV packages. Python is a dynamic multi-paradigm programming language with automatic memory management, supporting object-oriented, imperative, functional programming and procedural styles. One of the main advantages of using Python in application development is the existence of an extensive library of standard methods.

PIP is a package management system created to manage and install lists of software packages written in the Python programming language. In Fig. 1 and 2 the configuration of the PIP package management system in an Ubuntu Linux Machine is presented.

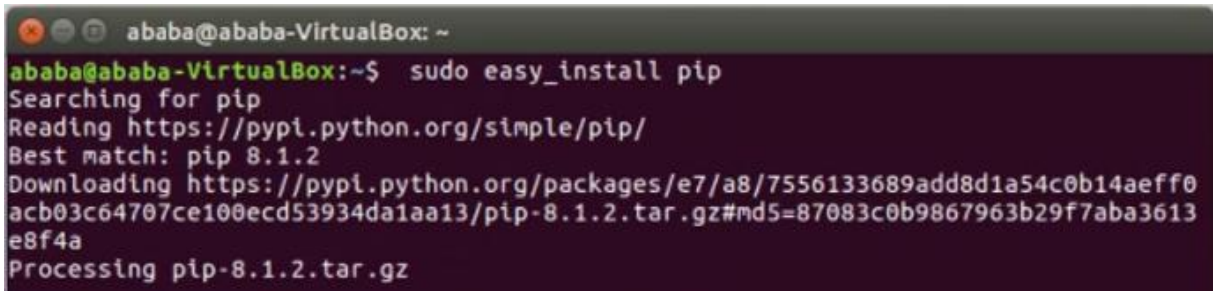

FIG. 1. *sudoeasy_install pip* command in Linux CLI

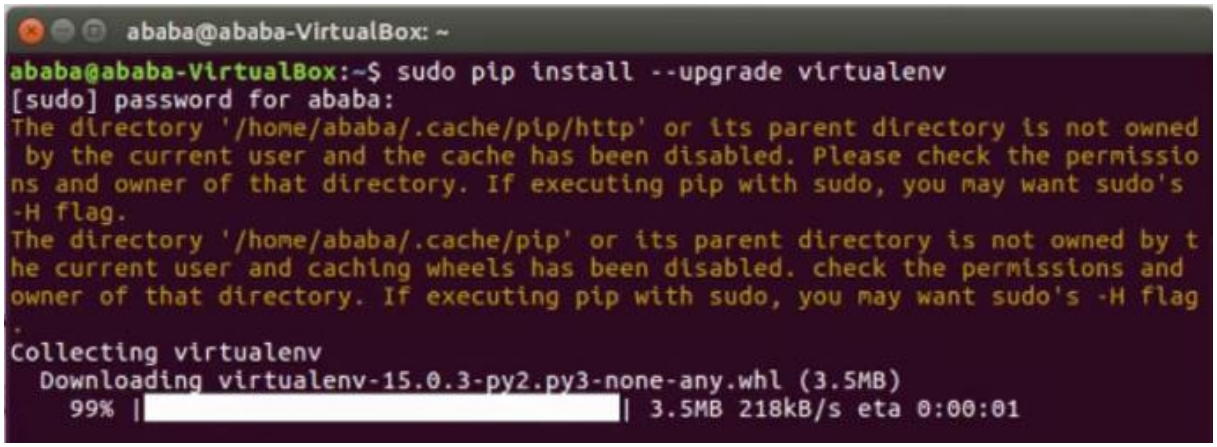

FIG. 2. *sudopip install --upgrade virtuallenv* command in Linux CLI

Once we have configured PIP on the machine, the needed packages for the OpenFace application can be integrated:

- NumPy is a Python extension able to add high-level mathematical functionality and support for large, multi-dimensional arrays and matrices. Furthermore it can be even used as a multi-dimensional container of generic data.
- SciPy is an open-source library for Python programming language used for technical and scientific computing and particularly used in mathematics and engineering. The SciPy library is used together with the NumPy extension to work with multidimensional arrays providing functions for linear algebra, random number generation or data transformation. SciPy also includes sub-packages like: *cluster, weave, sparse, ndimage, linalg, constants* or *fftpack*.
- Scikit-learn is an open source Python library used for machine learning and includes multiple algorithms for classification, clustering or regression with *support vector machines*, *k-means*, *random forests*, *DBSCAN* etc. The Scikit-learn library is built to work together with the NumPy extension and SciPy library for data mining and data analysis applications.
- Scikit-image is a Python library used for image processing and includes various algorithms for geometric transformations, filtering, segmentation, feature detection, color space manipulation and analysis. The Scikit-image library is built to work together with the NumPy extension and SciPy library and it is open-source.
- OpenCV is a cross-platform and open-source library, both for commercial or academic use, built to work with real time high-level acquiring, processing, analysing and understanding of digital videos or images. The OpenCV library offers support for the development of applications dedicated to: facial recognition, gesture recognition, human–computer interaction, motion tracking, augmented reality or emotion estimation.
- Torch is an open-source library and framework developed for deep machine learning providing various algorithms working with N-dimensional arrays, linear algebra and numeric optimization routines, neural network and energy-based models.

The video capture and real time media transfer is realized using WebRTC technology, a new standard defined by the IETF (Internet Engineering Task Force) and W3C (World Wide Web Consortium) to enable real time peer-to-peer communications between web browsers using a collection of common communication protocols.

WebRTC enables real-time video, audio or data communications between browsers without the need of additional plugins or modules that can make the experience intractable and complicated and can even generate security vulnerabilities because these plugins often need updates.

The access to the web camera is made trough the WebRTC built-in *MediaStream* API interface which was designed to grand easy access to media streams from local webcams or microphones using HTML5 and JavaScript objects.

Every *MediaStream* object includes multiple *MediaStreamTrack* objects which represents video and audio streams from different input devices. Each *MediaStreamTrack* object can include multiple channels (e.g. left and right audio channels).

When the user enters the application the web browser generates a default pop-up where the user is asked if he wants to grant the access to the webcam (Fig.3):

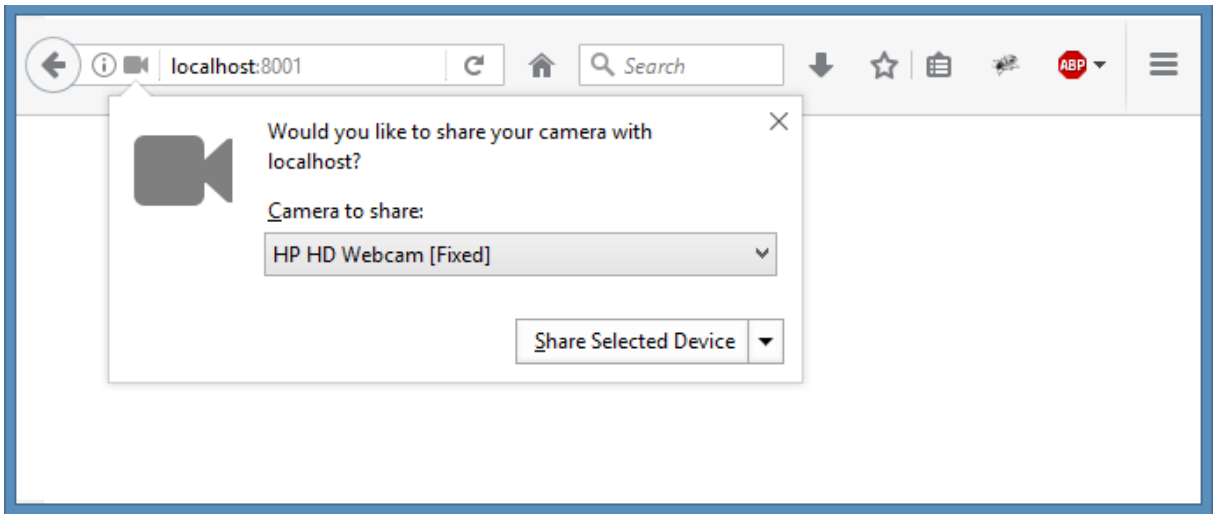

FIG. 3. Example of a web browser requesting acces to local webcam

Once the user has given consent to local web camera access, the web streaming is displayed in the browser and the face recognition analysis can begin, as presented in Fig.4.

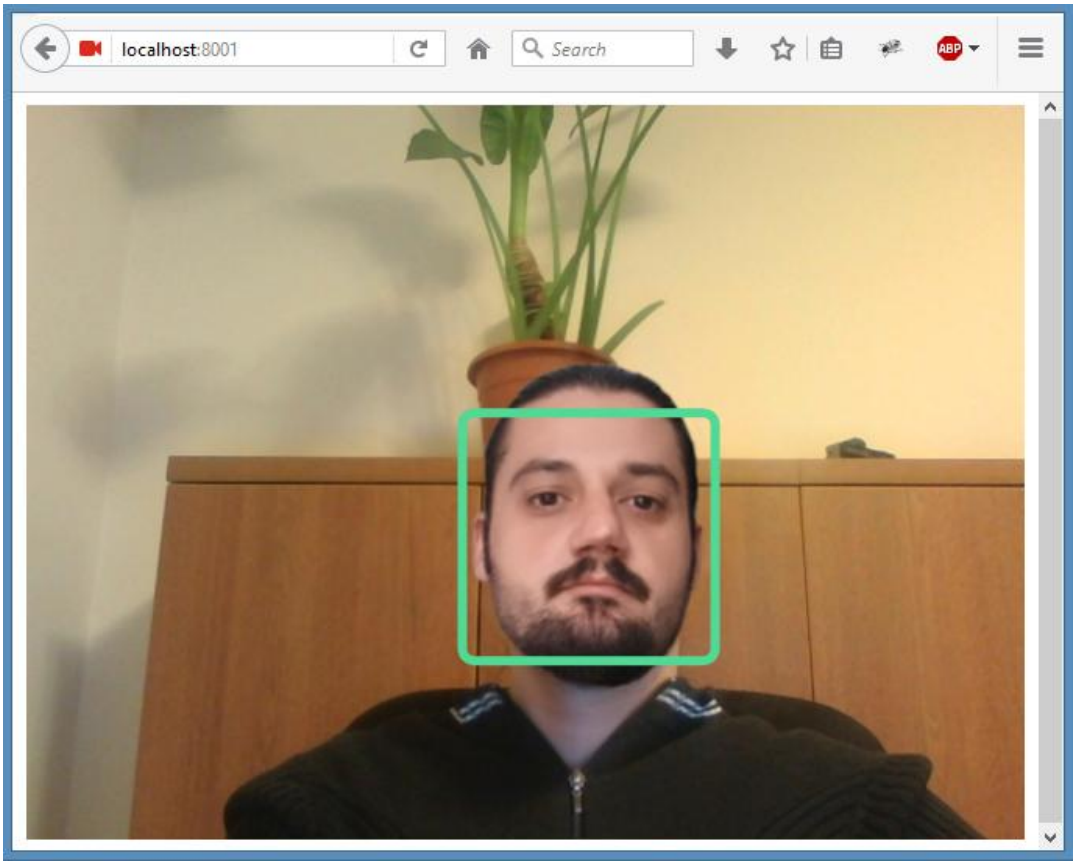

FIG. 4. Example of web streaming and face recognition analysis in the browser

The face recognition training dataset can be stored in the cloud and can be processed with TensorFlow open-source software library for machine learning or other deep learning neural networks systems and algorithms.

For the facial recognition the ID pictures of the students can be used or a set of images provided and validated by each student in the course/s enrolment. For best results it is recommended to have more pictures for each student in the training set.

3.2 Microsoft Azure Cognitive Services

Microsoft Azure is an array of integrated cloud services designed for building, deploying and managing applications and servers through a global network of data centers. Amongst the most important services provided are the: compute services (VMs, App Services, Websites and WebJobs), mobile services, storage services, machine learning and data management.

The Microsoft Azure Cognitive Services represent a set of key services designed to provide the tools needed to augment users experiences using compelling machine-based intelligence which includes powerful AI algorithms for: speech, vision and knowledge.

One of the most remarkable features of the Microsoft Azure Cognitive Services is the Face API.

Face API is a cloud-based service which uses the most advanced face algorithms and deep learning mechanisms in order to ensure two main functions: face detection with attributes and face recognition.

The possibilities arisen with this fairly new cloud-based service are virtually limitless stretching from basic face recognition for authentication to highly complex collective emotions analysis such as determining the impact of a speech based on the audience micro expressions or determining the target audience for a product, based on the attributes of the group which manifested most interest (age, gender or hair color).

From the educational use-case perspective, the Microsoft Face API can be used in developing:

- secure access systems able to grand conditional access to research laboratories
- VUE authentication systems
- systems which can monitor students presence to a course

One of the most remarkable advantage of the Microsoft Face API is that, besides actual face recognition, it can also provide valuable information regarding subject's emotions and can organize faces together into groups, based on their visual similarity. Having available this raw data, various statistical studies can be conducted which can offer important information about the impact of a course or can how can the University can improve a certain study program.

The process of developing and deploying such an application is relatively simple as it is all cloud-based. One must create a Microsoft Azure account and, from the dash-board, to select the type of application and the necessary recourses.

For demonstrations purposes, we have deployed such an application [8] on a Virtual Machine hosted by Microsoft Azure.

The main features of the implemented application are:

- face detection with
- emotion tracking and
- age approximation,
- face identification and
- face verification.

In Fig. 5, the face detection feature is presented. In both cases, the application also determines the subject's gender, emotion (smiling or not), facial hair presence or if the subject wears glasses.

Making use of the highly complex face algorithms, the application also approximates the subject's age, base on the face attributes.

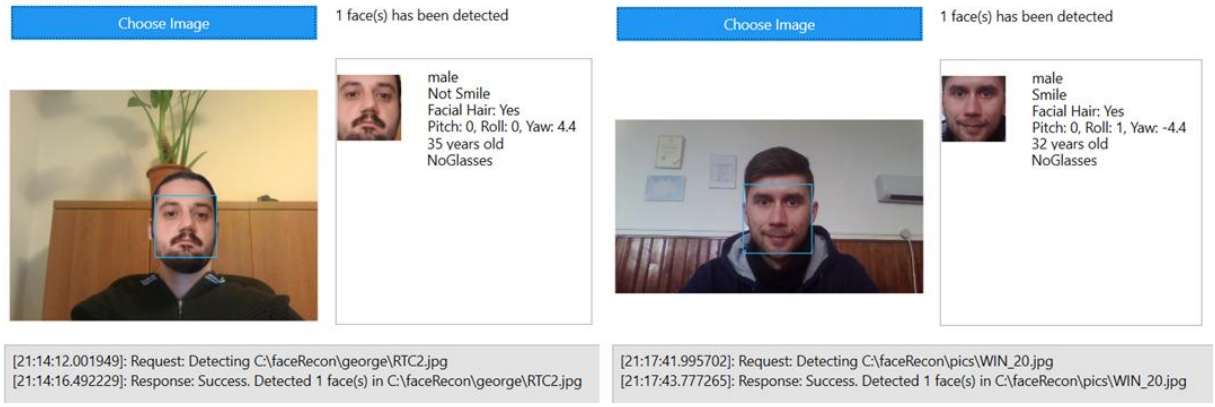

FIG. 5. Example of face detection with subjects not smiling and smiling.

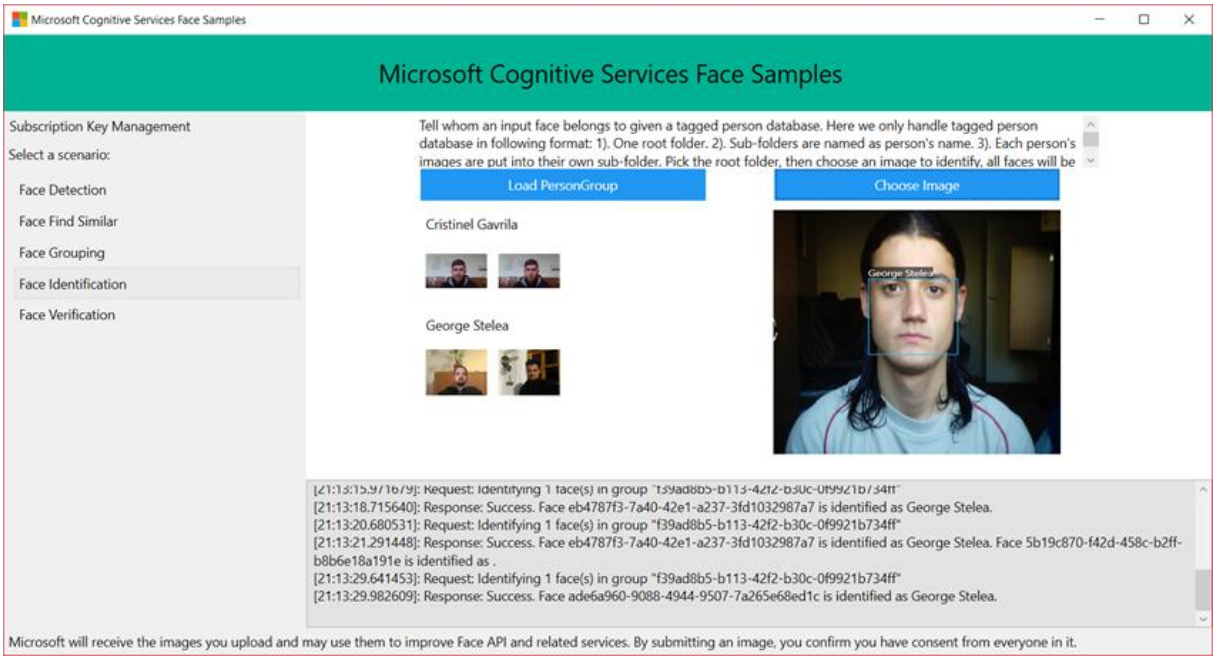

FIG. 6. Example of face identification.

Fig. 6 presents the face identification feature. For this feature the app requires some previous pictures which are use as a reference.

The versatility of the Microsoft Face API ensures that identification succeeds even if the subject pictures were taken in different time frames, as it is in the case presented in Fig. 6, where the subject's age in the reference picture is significantly higher than the one used for identification.

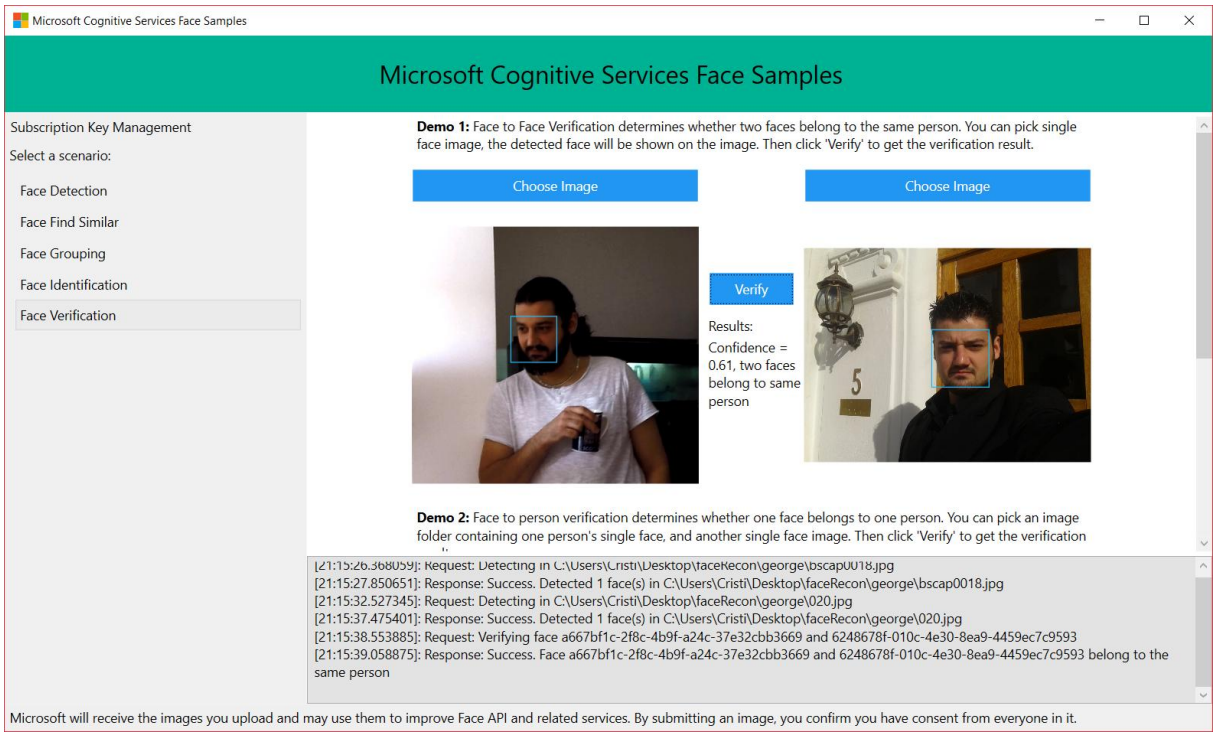

FIG. 7. Example of face identification with positive result.

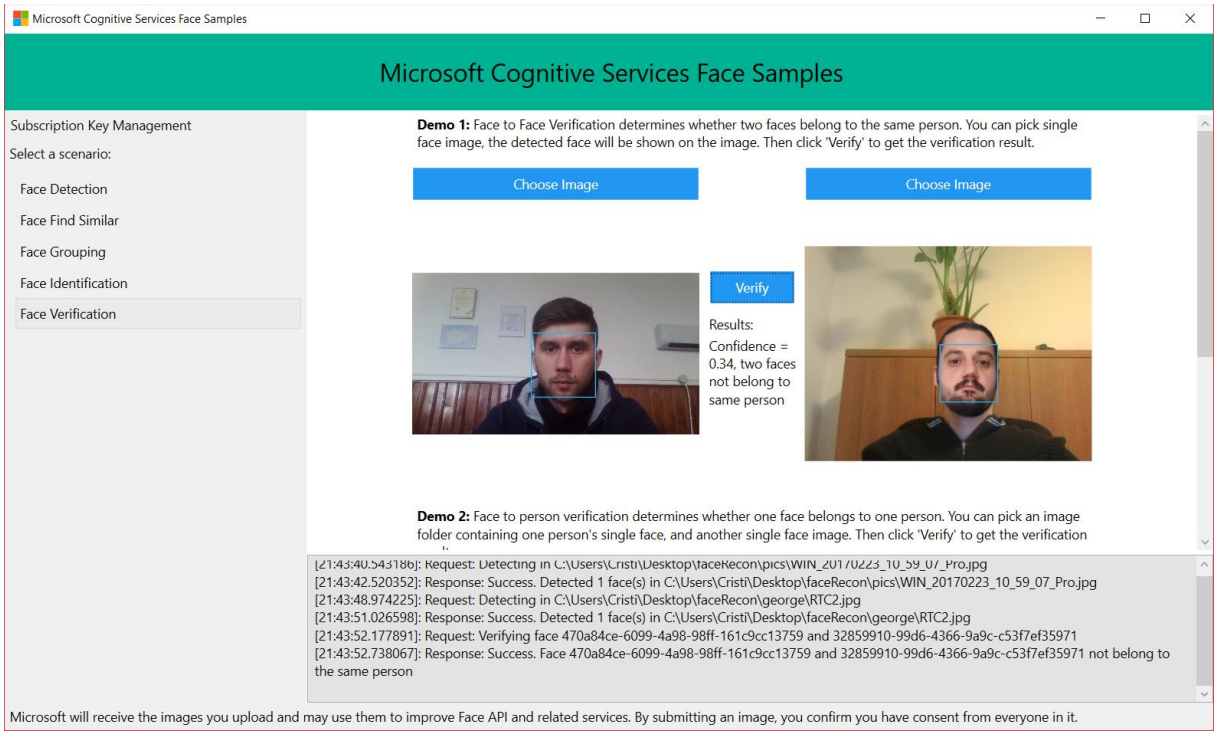

FIG. 8. Example of face identification with negative result.

Fig. 7 and 8 present the face identification feature which takes as input two pictures and applies the Face API algorithms and determines whether the same subject is in both or not.

For all the features, the application uses static images, but with no significant effort this can be changed into a live stream captured and transmitted to the VM by an IP camera. This way real-time analysis can be performed and highly accurate statistics can be provided depending on the targeted criteria.

IV. CONCLUSIONS

The solutions we have accomplished have a wide spectrum of applicability, from ensuring a secure access into a building, to monitoring group emotions in real-time all in a cloud-based environment. Furthermore they have the potential to enhance both human behavior and human interaction by analysing and understanding human emotions in different conditions. From the "Education as a Service" paradigm perspective, services such as this can ensure a propitious environment for online exams or for web seminars in aspects of authentication and increased interactivity and can also have a significant impact in the way the student-teacher interaction evolves by analysing the student behavior/interest (or lack of interest / lack of understanding) during a course – quantified by his/her face attributes.

We have demonstrated a face recognition cloud solution that can be integrated into a learning management system (LMS) and can be used both in university education as well as in other educational environments. The access policies to the educational resources can be restricted at the network level using SDN (Software Defined Networks) and/or LTE (Long Term Evolution – communications) -enabled methods such as advanced functions for traffic and access policies [9]. The cloud system can be extended for additional data mining after the face recognition process using deep learning neural networks algorithms to analyse various details. (e.g. if the student in sad or is smiling, or to analyze its attention and feedback on the course).

Reference Text and Citations

[1] John D. Kelleher, Brian Mac Namee, *Fundamentals of Machine Learning for Predictive Data Analytics,* MIT Press. 2016 - ISBN: 978-0262029445

[2] Stan Z. Li, Anil Jain, *Handbook of Face Recognition,* Springer, 2nd ed. 2011 - ISBN: 978-0857299314

[3] Salvatore Loreto, Simon Pietro Romano, *Real -Time Communication with WebRTC* , O'Reilly Media, Inc., 2014 - ISBN: 978-1-4493-7187-6;

[4] Shai Shalev-Shwartz, Shai Ben-David, *Understanding Machine Learning: From Theory to Algorithms,* Cambridge University Press , 2014 - ISBN: 9781107057135

[5] Ian Goodfellow, YoshuaBengio, Aaron Courville, *Deep Learning,* MIT Press, 2016 - ISBN: 978-0262035613

[6] Nishant Shukla, *Machine Learning with TensorFlow*, MEAP - Manning Publications, 2016 - ISBN: 9781617293870

[7] OpenFace - Free and open source face recognition with deep neural networks, https://cmusatyalab.github.io/openface/, Accessed. 2016

[8] https://github.com/Microsoft/Cognitive-Face-Windows, Accessed 2017

[9] Titus C.Balan, *Network policy function virtualization via SDN and packet processing* - Review of the Air Force Academy, Brasov – RO, 2014