Project Report

Problem Statement:

To design a face recognition system which is useful to patients to sign in:

a) at the hospital front staff management (FRS1) without waiting in line.

b) at the doctor cabin (FRS2) to help doctors providing patient's previous health records which is stored in hospital's database .

Providing previous health reports of the patients using a FRS2 in doctor's room , which will help doctor to understand patients health pattern.

Few reasons to implement FRS2 in hospital:

1. A patient might forget to carry some reports or might think that some reports are not necessary to report it to the doctor. But those reports might contain crucial information which might help doctor to consider them for his or present status regarding his/her health condition.

2. Patient might consume medicine on regular basis .Doctor would be able to know about that information through FRS2 as well and he prescribes the medicine for the present problem accordingly.

3. This will be helpful to patients to track their past medical reports and store them securely.

Introduction:

As we see in most of the hospitals, people tend to rush at the counter to fill the forms to get doctor's appointment . To overcome this problem , we can build an application which uses face recognition technology to provide better service. The reason for choosing face recognition technology over fingerprint scanners, iris scanners is that , there is possibility of infection transmission and impracticality especially when the patient is unconscious. Before getting into the details of the Face recognition system , lets understand what is face recognition?

A facial recognition methodology is a technology capable of identifying or verifying a person from digital image. Nowadays the security of person, information or assets is becoming more complex and it is considered as an issue. The crimes like credit card misuse and computer hacking or security breach in organizations are increasing day by day. The face recognition technology is nothing but the branch of biometrics which allows a person to be identified and verifiable data which are unique and specific to them. Details such as distance between eyes or shape of the chin are then converted into mathematical representation and compared to data on other faces collected in a face recognition database. The data about a particular face is often called a face template and is distinct from a photograph because it is designed to include certain details that can be used to distinguish one face from another.

Some face recognition systems, instead of positively identifying an unknown person, it is designed to calculate a probability match score between the unknown person and specific face templates stored in the database. These systems will offer with several potential matches, ranked in order of likelihood of correct identification, instead of just returning a single result. So, result is determined with an optimized one.

Methodology:

Neural network approach is more preferred among other approaches to design a face recognition system.

A neural network is a series of algorithms that endeavors to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates.

To build a face recognition system we use a pre-trained model called FaceNet.

FaceNet learns a neural network that encodes a face image into a vector of 128 numbers. By comparing two such vectors, you can then determine if two pictures are of the same person. The triplet loss is an effective loss function for training a neural network to learn an encoding of a face image.

To know more about FaceNet , below is the link of FaceNet research paper : https://arxiv.org/pdf/1503.03832.pdf

FaceNet architecture is shown below:

type	patch size/ stride	output size	depth	#1×1	#3×3 reduce	#3×3	#5×5 reduce	#5×5	pool proj	params	ops
convolution	7×7/2	112×112×64	1							2.7K	34M
max pool	3×3/2	56×56×64	0								
convolution	3×3/1	56×56×192	2		64	192				112K	360M
max pool	3×3/2	28×28×192	0								
inception (3a)		28×28×256	2	64	96	128	16	32	32	159K	128M
inception (3b)		28×28×480	2	128	128	192	32	96	64	380K	304M
max pool	3×3/2	14×14×480	0								
inception (4a)		14×14×512	2	192	96	208	16	48	64	364K	73M
inception (4b)		14×14×512	2	160	112	224	24	64	64	437K	88M
inception (4c)		14×14×512	2	128	128	256	24	64	64	463K	100M
inception (4d)		14×14×528	2	112	144	288	32	64	64	580K	119M
inception (4e)		14×14×832	2	256	160	320	32	128	128	840K	170M
max pool	3×3/2	7×7×832	0								
inception (5a)		7×7×832	2	256	160	320	32	128	128	1072K	54M
inception (5b)		7×7×1024	2	384	192	384	48	128	128	1388K	71M
avg pool	7×7/1	1×1×1024	0								
dropout (40%)		1×1×1024	0								
linear		1×1×1000	1							1000K	1M
softmax		1×1×1000	0								

Improvements :

FaceNet model takes the input image and outputs 128 encodings of that particular image. Output encodings is used further to improve the accuracy of the model. I have implemented two techniques to improve the accuracy of FRS, which are explained below.

1.One such technique is transfer learning. Transfer learning enables us to use the pre-trained models from other people by making some changes I.e. we can implement pre-trained models to accelerate our solutions.

But the problem of using this technique in my application is that, every time a new user is added the model has to be rebuilt. Consider a case of huge database, it takes a lot time to build a model. In this case, it reduces latency and we cannot compromise with the accuracy of the model as well. Below shown the summary of the model built which is used for confirmation of the person:

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
flatten_37 (Flatten)         (None, 128)               0
_________________________________________________________________
dense_115 (Dense)            (None, 128)               16512
_________________________________________________________________
batch_normalization_30 (Batc (None, 128)               512
_________________________________________________________________
dense_116 (Dense)            (None, 17)                2193
=================================================================
Total params: 19,217
Trainable params: 18,961
Non-trainable params: 256
_________________________________________________________________
```

This model is 78 percent accurate in verifying person's face. Accuracy can be further improved by changing the number of layers or neurons used in it.

2.Another way to improve the model is, the way we capture the face data features of a person. This model takes 40 images of a person, where the person is asked to move the head from left to right. The average of the 40 128-encodings is stored in the FR_database along with the id (Here, each person in the database has his own ID). When the person is trying to log in, the FRS takes 40 images of that unknown person and the rest process is mentioned below:

1. It divides those 40 images into 5 sets, each set consisting of 8 images.
2. Each set is mapped to the model to get 8 128-encodings and the average of those 8 128-encodings is computed and compared with the encodings stored in the FR_database .
3. We get a list of scores after comparison and we return the id of least score observation.
4. This process is repeated for each set and creation of ids is finally determined.
5. The ID with the maximum count reverts.

This method improves the accuracy of FRS.

Additional Features:

Liveness Detection:

A person might act unwisely to FRS by holding a picture of a person in front of FRS, so it is necessary to check the liveness of a person to ensure that same person is trying to access his/her account. Dataset used to build the model contains 1360 face images (680 real face images and 680 fake face images).

I have collected face data from 17 people I.e. 40 face images from each person. And manually created fake images using those 680 real face images. The output is a label, where 1 indicating that it is a real image and 0 indicating that it is a fake image.

Below shown the summary of the liveness convo net:

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_17 (Conv2D)           (None, 94, 94, 32)        896
_________________________________________________________________
max_pooling2d_12 (MaxPooling (None, 47, 47, 32)        0
_________________________________________________________________
conv2d_18 (Conv2D)           (None, 45, 45, 32)        9248
_________________________________________________________________
max_pooling2d_13 (MaxPooling (None, 22, 22, 32)        0
_________________________________________________________________
conv2d_19 (Conv2D)           (None, 20, 20, 64)        18496
_________________________________________________________________
max_pooling2d_14 (MaxPooling (None, 10, 10, 64)        0
_________________________________________________________________
conv2d_20 (Conv2D)           (None, 8, 8, 128)         73856
_________________________________________________________________
max_pooling2d_15 (MaxPooling (None, 4, 4, 128)         0
_________________________________________________________________
dropout_5 (Dropout)          (None, 4, 4, 128)         0
_________________________________________________________________
flatten_5 (Flatten)          (None, 2048)              0
_________________________________________________________________
dense_13 (Dense)             (None, 512)               1049088
_________________________________________________________________
dense_14 (Dense)             (None, 128)               65664
_________________________________________________________________
dense_15 (Dense)             (None, 2)                 258
=================================================================
Total params: 1,217,506
Trainable params: 1,217,506
Non-trainable params: 0
```

The test accuracy of the above model is 88.33 percent. This model is used in the FRS to detect the liveness of the person.

Initially when a person is trying to access his account then it checks for liveness of that person, if the output of the liveness convo model is 0 then, person is considered as a fake and access will be denied.

Conclusion:

Two techniques have been implemented using the encodings obtained from the FaceNet model. Advanced security features like liveness detection is also added. Further improvements can be achieved by playing around with the networks mentioned above.

References:

1. Florian Schroff, Dmitry Kalenichenko, James Philbin (2015). [FaceNet: A Unified Embedding for Face Recognition and Clustering](https://arxiv.org/pdf/1503.03832.pdf)
2. Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, Lior Wolf (2014). [DeepFace: Closing the gap to human-level performance in face verification](https://research.fb.com/wp-content/uploads/2016/11/deepface-closing-the-gap-to-human-level-performance-in-face-verification.pdf)