Super-Resolution using Deep Learning to Support Person Identification in Surveillance Video

Lamya Alkanhal[1], Deena Alotaibi[2], Nada Albrahim[3], Sara Alrayes[4], Ghaida Alshemali[5], Ouiem Bchir[6]
College of Computer and Information Sciences, Computer Science Department
King Saud University, Riyadh, Saudi Arabia

***Abstract*—Recently, video surveillance systems have been perceived as important technical tools that play a fundamental role in protecting people and assets. In particular, the recorded surveillance video sequences are used as evidence to solve violation, theft and criminal cases. Therefore, the identification of the person present on the crime scene becomes a critical task. In this paper, we proposed a Deep learning-based Super-Resolution system that aims to enhance the faces images captured from surveillance video in order to support suspect identification. The proposed system relies on an image processing technique called Super-Resolution that consists of recovering high-resolution images from low-resolution ones. More specifically, we used the Very-Deep Super-Resolution (VDSR) neural network to enhance the image quality. The proposed model was trained with CelebA faces dataset and used to enhance the resolution of the QMUL-SurvFace dataset. It yielded a Peak Signal-to-Noise Ratio (PSNR) improvement of 7% and Structural Similarity Index (SSIM) improvement of 3%. Most importantly, it increased the face recognition rate by 45.7%.**

Keywords*—*Deep learning; image processing; super-resolution; surveillance video

I. Introduction

Nowadays, video surveillance cameras are crucial for ensuring people's safety and security. In fact, they become an essential evidence for investigating security related cases. More specifically, the identification of the persons recorded in the surveillance video becomes decisive in solving such cases. However, the involved person identification is not always possible due to the low-resolution (LR) stored video frames.

The resolution that refers to the amount of details can be defined as the number of pixels per frame length unit [1]. Thus, the resolution increases with the number of pixels [2]. However, although surveillance cameras can record high-quality videos, they have limited storage space. Accordingly, the size of the recorded video is decreased, which leads to quality degradation. An obvious technical solution would be to increase the storage capacity, but this would be very expensive. In addition, the LR image could suffer from further quality degradation when the scene is crowded, or the suspect's face is far away from the camera since zooming in the image would degrade the quality of the region of interest. Furthermore, there are more factors that may affect the image quality like bad weather or the lightening conditions of the scene.

One of the ways to alleviate the problem of a low resolution (LR) image is to generate a high-resolution (HR) image from its corresponding single LR one by learning the relationship between them. This process is called Single Image Super-Resolution (SISR). The Super-Resolution (SR) problem is considered as an ill-posed problem since there are several ways to construct an HR image from a single LR one [3]. Example-based SR is a common class of such algorithm that uses a prior knowledge obtained by learning a huge dataset of LR-HR image pairs [4], then use this knowledge to predict fine details in real-world images. Recently Convolutional Neural Networks (CNNs) have been proposed as a solution to solve the Super-Resolution problem [5][6][7].

In this paper, we improve the performance of the LR faces images captured from surveillance recorded videos by adopting a Very-Deep Super-Resolution (VDSR) convolutional neural network-based system [8]. The proposed system is trained to learn the estimation of the difference between an HR image and an LR image. This difference called residual image [9] is added to the LR one to obtain the final HR image.

II. Related Works

Recently, SISR systems using deep learning for both generic images and faces images enhancement have been reported in the literature [4][10] [11].

A. SISR for Generic Images using Deep Learning

The authors in [6], proposed an SR system called Super-Resolution Convolutional Neural Network (SRCNN). The input of this approach is a single LR image up-sampled using bicubic interpolation. The input image is then converted to its Y-channel. The system directly produces the HR image. The deep learning architecture consists of three convolutional layers. More specifically, the feature extraction is applied in the first convolutional layer. The second layer applies a fully connected non-linear operation, which maps feature maps to HR patches. The last layer merges the predictions to eventually produce the HR image. Fig. 1 illustrates SRCNN architecture where f_s are the filters sizes.

In the Efficient Sub-Pixel Convolutional Neural Network (ESPCN) approach [12], an HR image is convolved using Gaussian filter and downscaled by a factor r to produce an LR image, which is used as an input to the network. The system architecture consists of three hidden layers. The first two layers are convolutional layers for extracting features and the last layer is a sub-pixel convolution layer in which the LR image is up-sampled by a factor r via a pixel shuffle operation that rearranges pixels to produce HR output. The pixel shuffle operation converts $r^2\,(height \times width \times channel)$ LR feature representation to a $(r\,height \times r\,width \times channel)$

HR image as depicted in Fig. 2. An upscaling factor of 3 $(r = 3)$, is used such that for every pixel in the LR image there are 3×3 pixels (one pixel from each channel) in the HR image. Unlike SRCNN [6], the only upscaling occurs in the last layer, so the authors could extract the feature maps in LR space using small filters. Therefore, the number of computations can be reduced, and eventually, real-time performance can be achieved.

In [13], the authors designed an SR system called Orientation-aware Deep Neural Network for Real Image Super-Resolution. The system uses as input real-world captured LR image after converting it to its corresponding Y-channel. The network learns to output a residual image. The architecture consists of three layers CNN and same as [12] and has a pixel-shuffle operation. Fig. 3 displays the network architecture. In Fig. 3, I^{LR} and H^{HR} denote the LR and HR images, respectively. At the input layer, an operation called de-pixel shuffle is performed on the input, it arranges the image by converting its size from $height \times width \times channel$ to $\frac{height}{r} \times \frac{width}{r} \times r^2 channel$, where r is the scaling factor. This operation reduces the image size and increases the number of channels in order to accelerate the speed of the network. Convolutional layers extract three orientation-aware features by using horizontal, vertical, and diagonal kernels. After the first convolutional layer, there are 16 orientation-aware feature extraction and channel-attention modules (OAMs) that are used to fuse the orientation-aware features to produce more distinctive features that are used for LR-HR mapping. At Each OAM, a local residual unit is used to accelerate the training process. OAM architecture is shown in Fig. 4, F^{ver}, F^{hor} and F^{dia} denote the extracted features, and F^{fuse} is the fused features, while F^{CA} is the features after being enhanced by the attention channel. At the end of the network, a pixel shuffle operation is performed at the final output to reconstruct the HR image.

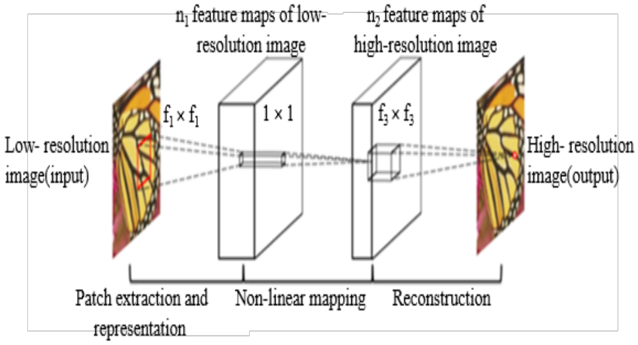
n_1 feature maps of low-resolution image
n_2 feature maps of high-resolution image
$f_1 \times f_1$
1×1
$f_3 \times f_3$
Low- resolution image(input)
High- resolution image(output)
Patch extraction and representation
Non-linear mapping
Reconstruction

Fig. 1. The Architecture of SRCNN [6].

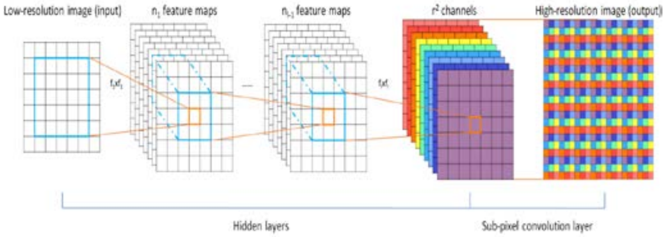

Fig. 2. The Architecture of ESPCN [12].

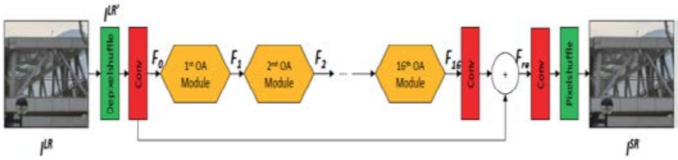

Fig. 3. The Architecture of Orientation-aware Deep Neural Network [13].

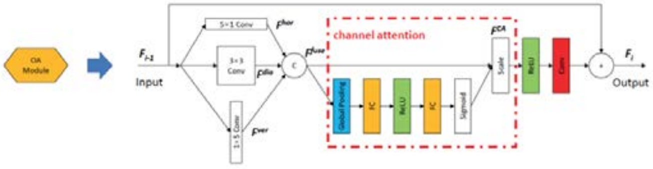

Fig. 4. The Architecture of OAM [13].

The authors in [14] introduced a system called Deep CNN with Skip Connection and Network in Network (DCSCN). Similarly, to [6] and [13], the network processes the original image's Y-channel and produces channels of corner pixels of each up-sampled pixel named 4ch (square of the scale factor channels). As displayed in Fig. 5, the network consists of two sub-networks: feature extraction network and reconstruction network. The feature extraction network receives as input image's pixels and extracts features of each pixel in seven CNN layers. Each layer is connected to the next layer and the reconstruction network by Skip connections. So, each layer's output is sent to the next layer and the reconstruction network simultaneously. As the reconstruction network receives extracted features, three parallelized CNNs (Network in Network) up-sample and reconstruct the image features and pass the output to the last CNN layer. This last layer outputs 4 channel image which is reshaped to construct an intermediate HR image. At the end, the intermediate HR image is added to the bicubic up-sampled input image to obtain the final HR image.

In [15] the authors proposed the Deep Recursive Residual Network (DRRN) network. The input is an LR image, which is generated using bicubic interpolation to the color components of the image. The deep neural network model learns to output the residual image. For this purpose, the proposed system architecture consists of 52 convolutional layers. Moreover, it has a recursive mechanism that contains two residual units. The first one is a global residual, which estimates the HR image from combining the input and the residual of the network. The second one is a local residual. It is located at every few stacked layers. Fig. 6 shows DRRN architecture.

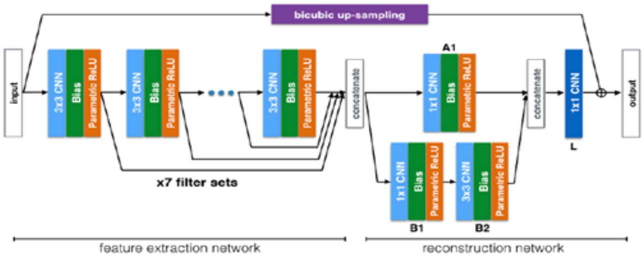

Fig. 5. The Architecture of DCSCN [14].

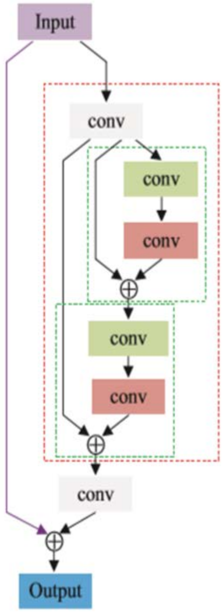

Fig. 6. The Architecture of DRRN [15].

In [8], the authors proposed a Very Deep Super-Resolution Convolutional Networks (VDSR). The input is an interpolated low-resolution (ILR) image. It is produced by extracting the Y channel from the HR image, then down-sampling it by bicubic interpolation, after that, the obtained LR image is up-sampled to match the original size of its HR. The network's output is the residual image. The network consists of 20 sequential convolutional layers which means it uses large information for HR image reconstruction (large receptive field). Also, each hidden layer has 64 filters of size 3×3×64 for feature extraction. The network architecture is shown in Fig. 7.

B. SISR for Face Images using Deep Learning

The authors in [16] proposed the Multi-Scale Competitive Convolutional Neural Network. It is a three layers CNN. The HR image is down-sampled then up-sampled using bicubic interpolation to obtain an LR image which is conveyed as input to the deep neural network. The system learns to output the residual image. Fig. 8 shows the system architecture. The first two layers have three parallel competitive multi-scale filters. These filters provide high contextual information for the SR image. Each layer produces three groups of features. These groups are then arranged to obtain non-overlapping groups. Then, the maxout activation function is applied to select the input for the next layer. The last layer is responsible for reconstructing the final image. This network uses faces images as training examples as well as generic images.

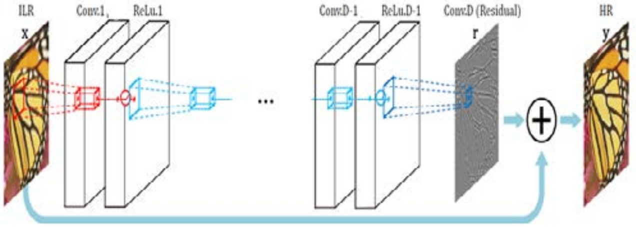

Fig. 7. The Architecture of VDSR [8].

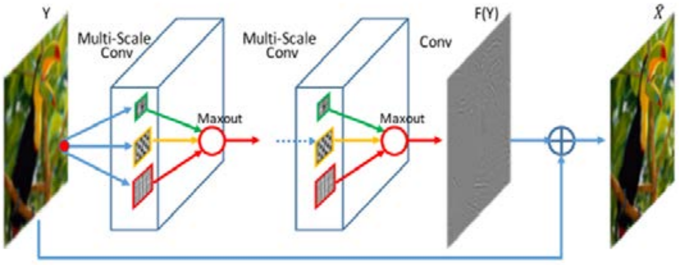

Fig. 8. The Architecture of the Multi-Scale Competitive Convolutional Neural Network [16].

The authors in [17] suggested the Attribute Augmented Convolutional Neural Network (AACNN) system to solve the SR problem. The input of this system is a down-sampled LR image of size 14×12 along with a feature attribute vector. As for the output, it produces an HR image with a size of 112×96. The architecture of the system consists of two sub-networks as shown in Fig. 9. They are generator and feature-extraction networks. The feature extraction network consists of two sub-branches A and B. A is responsible for extracting the fine-grained feature from the LR input image using three convolution layers. On the hand, B takes as an input the feature attribute vector with a dimension of 1×1×38 and expands its dimension from 38 to 504 using a fully connected layer. It is then reshaped to 14×12×3 to match the size of the LR image. The next convolution layers work as branch A. Then, branch A and B are both combined as one image to be fed to the generator network. In other words, the generator network learns the mapping between the LR and HR images through the features that it receives from the feature extraction. This helps the generator get a clearer vision of the HR image while up-sampling the image through the network using deconvolution layers. This yields the generation of the final HR image with a size of 112×96.

Unlike the system presented in [17], which is fully parallel, the authors in [18] proposed a partially parallelized network called Bi-channel CNN. The network receives a 48×48 bicubic interpolated LR image as input and learns to output an intermediate image reconstructed of the extracted face features. Fig. 10 represents the proposed architecture. Same as in [17], the network consists of two sub-networks: feature extractor and image generator. Feature extractor extracts input face features with three convolutional layers. The output is passed to the image generator which contains four fully connected parallelized layers in two groups. The first group reconstructs the image's features to produce an intermediate image, while the second group predicts the fusion coefficient α which controls the integration of the two channels of information. They are the up-sampled input image and the intermediate image.

As reported above, various SR approaches based on deep learning have been proposed in the literature. Some of these approaches are designed for generic image SR [6, 12, 13, 14, 15] while others are designed for the specific application of faces images SR enhancement [16, 17, 18]. In the following, we discuss the reported approaches in terms of context, convergence, and scale factor.

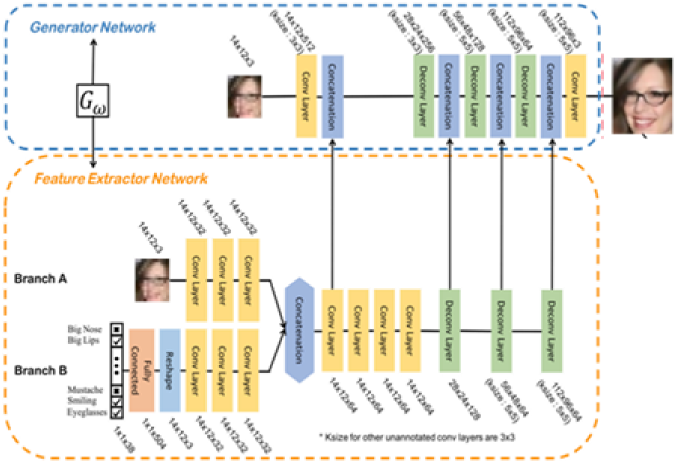

Fig. 9. The Architecture of AACNN [17].

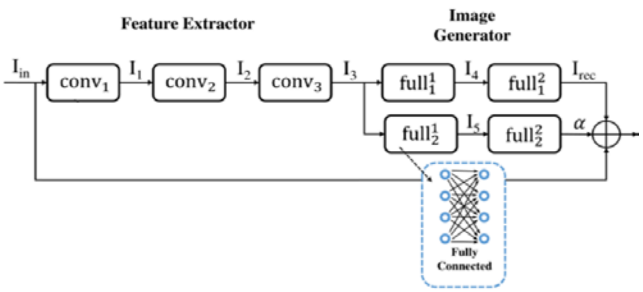

Fig. 10. The Architecture of Bi-Channel CNN [18].

Context: Most large depth networks use a large amount of contextual information. This allows retrieving more image details in order to obtain the HR image. The network in [6] relies only on the context of small regions of the image, which is inefficient for obtaining details of large-scale image. On the other hand, the works reported in [8, 15] use a large amount of contextual information allowing the reconstruction of HR images with more details.

Convergence: Since the LR image and the HR image are similar, learning the residual instead of learning the HR image reduces the required computations. The works reported in [8, 13, 14, 15, 16] learn residual images which speeds-up the training process.

Scale-factor: for real-world applications, image scales differ naturally depending on the application. However, most reported approaches consider only a single specified scale. On the other hand, the work in [8] proposed a system that handles multi-scale images for more realistic model.

A summary of the related works is presented in Table I. It compares the different approaches based on these characteristics: The preprocessing process applied to the input, the input image type, the proposed architecture, the training dataset, and the performance results. The performance of each network is measured using Peak Signal-To-Noise Ratio (PSNR) [19] or Structural Similarity Index (SSIM) [20]. As reported in Table I, the performance of the reported approaches shows that there are still ways of improvement.

III. Very Deep Super Resolution Network Structure

The Very Deep Super Resolution, VDSR, is a Convolutional Neural Network (CNN). It is designed to solve the SR problem for generic images [8]. The network structure consists of an input layer, twenty cascading pairs of hidden layers and an output layer. Each pair consists of a convolutional layer followed by a rectified linear unit. The network structure is illustrated in Fig. 11.

A. Input Layer

It is the first layer. It takes as an input an interpolated low-resolution (ILR) image with only one channel since the network is trained using the luminance channel of the input. The layer operates on the input. The size of the patches depends on the network receptive field which size is equal to the input image, in order to let the field views all the high features of the image [9].

B. Hidden Layers

The image input layer is followed by 19 alternating convolutional and rectified linear unit (ReLU) layers. The convolutional layer contains 64 filters each of size 3×3×64. They are applied to 3×3 regions across 64 channels of the input to extract features. The units (neurons) in the first convolutional layer are connected to local regions of the input image, and so on, each unit in convolutional layer m is connected to a subset of units of convolutional layer $m-1$. Thus, each unit has its own receptive field with respect to its input. This design implies that the learned filters of the network generate the highest response to a local input feature.

When the number of sequential layers is large, the filters by time become global, which leads to increase the size of the receptive field by 2 in height and width, the size is calculated by the formula: $(2D+1)(2D+1)$, where D is the number of the convolutional layer. In VDSR, there are 20 convolutional layers, so the receptive field (as well as the size of the patch) is 41 by 41. In SR, a large receptive field means large contextual information is taken into consideration when predicting image details [8].

C. Output Layer

As for the final layers, the penultimate layer is a convolutional layer, it has only one filter of size 3×3×64. The convolutional layer is followed by the regression unit.

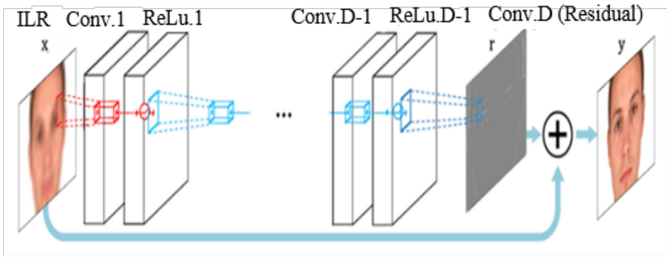

Fig. 11. VDSR Network Structure.

TABLE I. A BRIEF SUMMARY OF THE RELATED WORKS

Reference	**Preprocessing**	**Image Type**	**Architecture**	**Number of Layers**	**Training Datasets**	**Results**
[6]	Bicubic interpolation and Y channel extraction	Generic	CNN	3 convolutional layers	ImageNet [25]	PSNR = 29
[12]	Gaussian filter and down sampling	Generic	CNN with one sub-pixel convolutional layer.	3 convolutional layers	ImageNet [25]	PSNR = 29.495
[13]	Y channel extraction	Generic	CNN, 16 channel attention modules, pixel shuffle and de-pixel shuffle layer.	3 conventional layers	NTIRE2019 Real Super-Resolution challenge [26]	PSNR= 29.35 SSIM= 0.8599
[14]	Y channel extraction	Generic	Partially parallel Deep CNN with Skip Connection	11 conventional layers	Berkeley Segmentation Dataset [27]	PSNR = 37.02
[15]	Bicubic interpolation	Generic	DRRN with Several recursive blocks, followed by a convolutional layer	52 convolutional layers	Berkeley Segmentation Dataset [27]	PSNR = 28.14 SSIM = 0.788
[8]	Bicubic interpolation and Y channel extraction	Generic	CNN	20 conventional layers	Dataset in [28] and Berkeley Segmentation Dataset [27]	PSNR = 37.53 SSIM = 0.9587
[16]	Bicubic interpolation and Y channel extraction	Faces	Parallel competitive multi-scale filters CNN	3 conventional layers	Dataset in [29] and Berkeley Segmentation Dataset [27]	PSNR = 32.42 SSIM = 0.7808
[17]	Down sampling	Faces	2 Parallel CNN sub-networks	12 conventional layers	CelebA [23]	PSNR= 27.4007 SSIM= 0.8036
[18]	Bicubic interpolation	Faces	Partially parallel Bi-channel CNN	3 conventional layers and 4 fully connected layers	Their own [18]	PSNR = 34.63 SSIM = 0.92

IV. SUPER-RESOLUTION USING DEEP LEARNING TO SUPPORT PERSON IDENTIFCATION IN SURVEILLANCE VIDEO

SISR is considered a critical issue in many applications that deal with LR images and need to extract high information details from them. Face recognition from surveillance camera records is one of the applications that suffer from such a problem, thus making the task of identifying the persons present in the scene difficult. For this purpose, we propose a SISR technique that uses Very-Deep Super-Resolution (VDSR) system to recover HR images from LR images [8]. VDSR has been used in [8] with generic images while the focus of this paper is on faces images captured by surveillance videos. Conveying the obtained HR images to a face recognition system would enhance the performance of the recognition system since the HR image contains more details.

As reported in the literature, CNN has been successfully used to learn HR image from LR image [6]. However, CNN based approaches have some drawbacks. First, they only preserve the contextual information over small image regions. Second, the training process is very long since the input image has to go through all layers until it reaches the output layer. Third, the systems are trained using a unique image scale, which makes them incapable of handling images of different scales.

In order to alleviate the previously mentioned drawbacks, we propose to use the VDSR architecture [8] to learn the residual image. It is a very deep architecture that consists of twenty sequential layers. Furthermore, it organizes many adjacent small filters over the input image that preserves the contextual information over large regions of the image. These proprieties would yield a better SR system. In fact, since it uses a large receptive field, it considers the contextual details spread over large regions of the input. This increases the amount of context a neuron can see from the input, to map it to the output [21]. Moreover, the proposed system learns the residual image. This yields its convergence in a reasonable time compared to the systems that learn the HR image directly. In addition, it has been shown in [8] that the performance of the residual network is higher than the non-residual one. Furthermore, the proposed system uses a single network for training multi-scaled data. Furthermore, it has been empirically shown in [8] that the performance of the network increases with its depth. In other words, very deep CNN improves SR system performance.

A. Training Framework

For training the VDSR network, the first step is processing the training images. The images are HR, they are transformed into YCbCr space, and then, the luminance channels (Y) of them are extracted. After that, the images are down-sampled using different scale factors to obtain LR images. The LR images are resized by bicubic interpolation [22] to make them match the original sizes of their HR images. Lastly, the residual images are calculated, and they are stored along with the resized images (LR images) [9].

When the training phase is initiated, the network's weights (filters) values are randomly generated and the biases are initialized to zeros. The network learns to output the estimated residual image of the input by minimizing the loss function which is the mean squared error. This function is calculated at the last layer and is defined as.

$$L = \frac{1}{2}\|r - f(x)\|^2 \quad (1)$$

where r is the residual image and (x) is the network predicted output. The network is able to minimize this function by mini-batch gradient descent that uses the backpropagation method [8]. In addition, the network is trained using multiple scale images. This is achieved by dividing the image into non-overlapping sub-images and having an input patch size equal to the receptive field. One mini-batch contains 64 sub-images of the same scale [8].

V. Experiments

A. Datasets Description

CelebA [23] and QMUL-SurvFace [24] benchmark datasets are used to train and assess the performance of the proposed system. CelebA [23] is a faces dataset for celebrity faces images. It includes 10,177 of identities and 202,599 of face images. The dataset covers different backgrounds with pose variations. As for the QMUL-SurvFace benchmark dataset [24], it includes large scale surveillance faces images. It has been introduced as a benchmark for the surveillance face recognition challenge 2019. This benchmark contains 463,507 faces images of 15,573 different persons captured in various real-world scenes with large space and time variations. The set of images are LR images by nature. In other words, they are not the result of down-sampling HR images.

As the faces images of CelebA dataset [23] are HR, the dataset is used for training and testing the VDSR network. However, QMUL-SurvFace [24] dataset cannot be used to train the network since it has only LR faces images. Instead, it is used to test the trained system with a face recognition system [30] as an evaluation tool.

B. Experimental Setting

A network depth of 20 and training batches of size 64 are used as in [8]. Moreover, the Momentum parameter is set to 0.9 and the decay parameter to 0.0001. Moreover, the number of epochs is determined by early stopping technique, where a validate set is tested by the model after a certain number of epochs to check for overfitting occurrence and stop the training then. The learning rate is set to 0.1. In addition, we use a zero-padding technique.

C. Experiment 1

In this experiment, we trained the VDSR system using a subset of the first 110,000 images from CelebA dataset [23]. As a preprocessing step, CelebA dataset images were cropped to get rid of the background and include only the face. The images were originally of size 218×178 and became 116×105. The dataset was randomly split into a training set of 66,000 images and test set of 44,000 images. The VDSR was trained to handle three scale factors, 2, 3 and 4. So, the training images were down-sampled by these scale factors, then up-sampled by bicubic interpolation [22] and fed to the model.

The training set was provided as input to the VDSR system to train the network. The model was trained over 5 epochs. Once the training phase was finished, the test set was conveyed as input. PSNR [19] and SSIM [20] are used to compare the obtained HR images to the target ones, the training and testing results with respect to each scale factor are reported in Table II and Table III, respectively. We can see that the model's results outperform the bicubic interpolation's results [22]. Also, we can observe from the tables that as the scale factor is smaller as the performance results are better. It can be explained by the fact that the SR problem is simpler as the down-sampling factor is small.

Fig. 12 and Fig. 13 display the performance comparison of PSNR and SSIM, respectively. They show a comparison between the testing results of the trained VDSR model using CelebA, the training results of the trained model using CelebA, the testing results for bicubic interpolation, and the training results for bicubic interpolation.

TABLE II. Performance of VDSR Model and Bicubic Interpolation on the Training Set

Scale factor	VDSR Model		Bicubic Interpolation	
	PSNR	***SSIM***	***PSNR***	***SSIM***
×2	32.426770	0.971939	30.259453	0.957807
×3	27.726198	0.934817	26.064288	0.913828
×4	27.553009	0.923985	25.706819	0.895698

TABLE III. Performance of VDSR Model and Bicubic Interpolation on the Test Set

Scale factor	VDSR Model		Bicubic Interpolation	
	PSNR	***SSIM***	***PSNR***	***SSIM***
×2	32.417566	0.971969	30.259927	0.957875
×3	27.729268	0.934914	26.070009	0.913938
×4	27.551795	0.924075	25.711985	0.895859

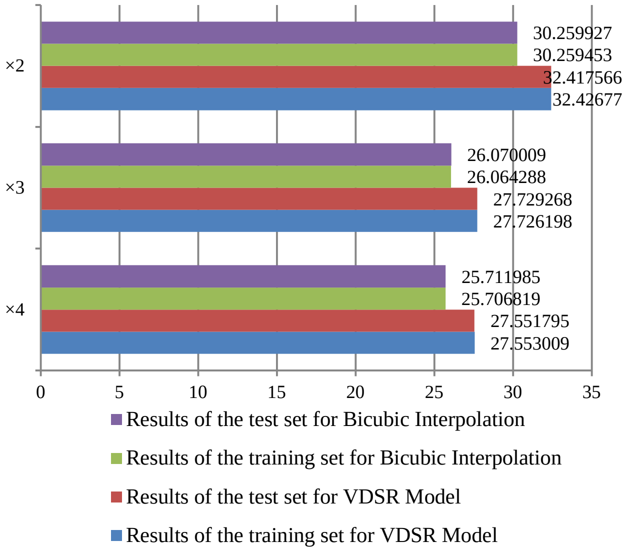

Fig. 12. PSNR Performance Comparison between the Training and Testing Results of VDSR Model and Bicibic Interpolation.

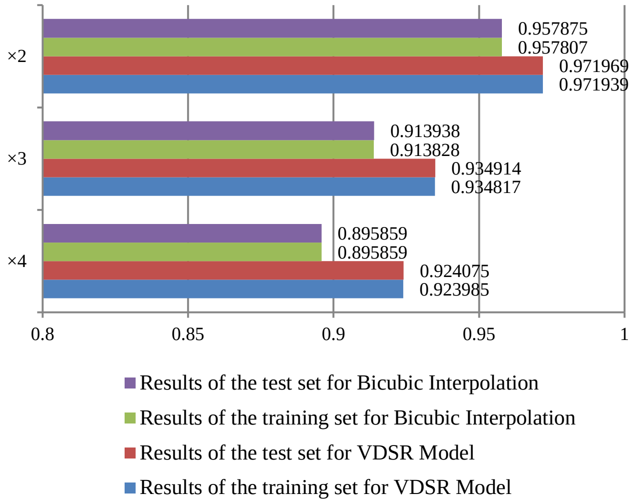

Fig. 13. SSIM Performance Comparison between the Training and Testing Results of VDSR Model and Bicibic Interpolation.

Fig. 14 shows a sample image from CelebA. Fig. 15, Fig. 16 and Fig. 17 display the results of the SR enhancement using the multi-scaled VDSR model with respect to scale factors 2, 3, and 4. For each considered scale factor, the LR image obtained by down-sampling the original image by the corresponding scale factor, the learned residual image, and the resulting HR image are displayed.

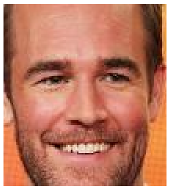

Fig. 14. Sample Cropped Image from CelebA.

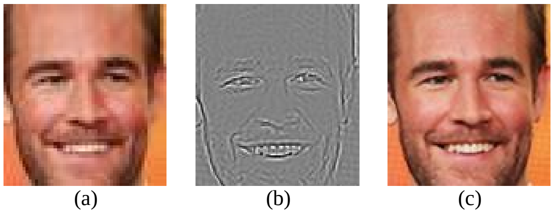

Fig. 15. Results of the SR Enhancement using the Multi-Scaled VDSR Model with Respect to Scale Factor 2 on the Sample Cropped Image from CelebA . (a) LR Image Obtained by Down-Sampling the Original Image by a Scale Factor of 2, (b) the Learned Residual Image, and (c) the Resulting HR Image.

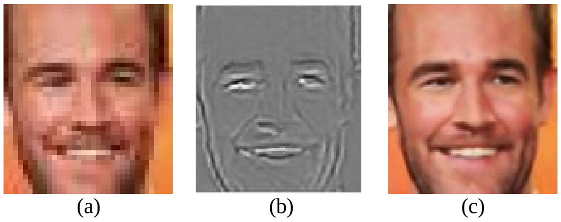

Fig. 16. Results of the SR Enhancement using the Multi-Scaled VDSR Model with Respect to Scale Factor 3 on the Sample Cropped Image from CelebA . (a) LR Image Obtained by Down-Sampling the Original Image by a Scale Factor of 3, (b) the Learned Residual Image, and (c) the Resulting HR Image.

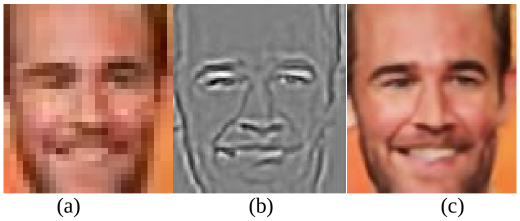

Fig. 17. Results of the SR Enhancement using the Multi-Scaled VDSR Model with Respect to Scale Factor 4 on the Sample Cropped Image from CelebA . (a) LR Image Obtained by Down-Sampling the Original Image by a Scale Factor of 4, (b) the Learned Residual Image, and (c) the Resulting HR Image.

D. *Experiment 2*

This experiment is conducted on the QMUL-SurvFace benchmark dataset [24]. Since this dataset has only LR face images, it cannot be used to train the VDSR. Instead, it was used to test the trained system using CelebA dataset [23] obtained from experiment 1. In other words, using the model obtained after training the network in experiment 1, the LR images from the QMUL-SurvFace benchmark dataset [24] were conveyed as inputs. The learned residual images were added to their corresponding LR ones. Since the target HR images were not available, we used a different assessment approach. More specifically, we used a face recognition system [30] as an evaluation tool. We provided LR faces images to the recognition system and recorded the performance of the system. Then, we conveyed the learned HR ones and computed the performance of the recognition system. Finally, we compared the two results.

We should mention here that the face recognition system was just used as an assessment tool and that the recognition task is out of the scope of this paper. This implies that the performance of the recognition system was not important but rather the improvement of the performance (if any) between the two considered scenarios. For this purpose, we chose a simple face recognition system proposed in [30]. It extracts the Histogram of oriented gradients (HOG) feature [31] from the faces images. The obtained visual descriptors are classified using the K-nearest neighbors (KNN) [32] with a number of neighbors equal to 1 (K=1). The distance between the HOG features of the faces in the dataset is calculated using the Euclidean distance.

We used 1,355 identities from QMUL-SurvFace dataset [24] for training and testing the recognition system. To match the recognition system requirements, 41 images (the average number of images per identity) for each identity were picked randomly as it requires a fixed number of images. As for the validation technique, leave one out cross-validation was used. The recognition system was tested with the original QMUL-SurvFace dataset images and the enhanced images using the VDSR model trained in experiment 1. Finally, the test results were compared based on the ratio of the correctly recognized face images.

The recognition system was tested with three different scales 2, 3, and 4. Table IV reports the performance of the recognition system with respect to the considered scale factors. Fig. 18 shows the percentage of performance enhancement when using the learned SR images instead of the original LR images. From Fig. 18, we notice that the performance of the recognition system improved when using the enhanced SR images instead of the original LR one. The increase in performance is almost the same for scale factors 2, 3, and 4 (around 45.5%).

TABLE IV. PERFORMANCE OF THE RECOGNITION SYSTEM WHEN USING LR IMAGES, AND WHEN USING SR IMAGES WITH RESPECT TO SCALE FACTORS 2, 3, AND 4

	Ratio of correctly recognized faces	**Percentage of performance improvement**
LR images	0.359	0
HR images using scale 2	0.523	45.68%
HR images using scale 3	0.522	45.40%
HR images using scale 4	0.523	45.68%

Fig. 18 shows the percentage of performance enhancement when using the learned SR images instead of the original LR images.

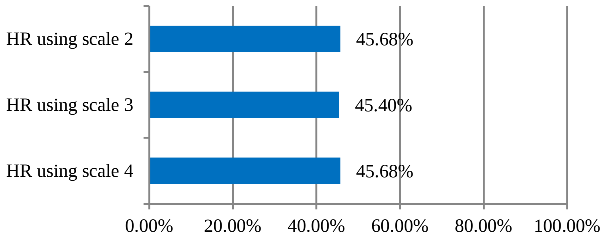

Fig. 18. Percentage of Performance Enhancement when using the Learned SR Images Instead of the Original LR Images with Respect to Scale Factors 2, 3 and 4.

VI. CONCLUSIONS AND FUTURE WORKS

In this paper, we proposed to generate a high-resolution (HR) face image from low-resolution (LR) one for the purpose of person recognition from surveillance camera records. For this purpose, we developed a Deep learning-based SISR system technique that uses Very-Deep Super-Resolution (VDSR) architecture. It is a network with a large number of layers that learns the residual between HR and LR images and accepts images of different scales. The conducted experiments showed that the VDSR model enhanced the face images resolution. In addition, the performance of the recognition system inferred that the proposed VDSR system improves the recognition rate by a ratio of 45.6%.

As future works, we intend to investigate single-scaled models and check the performance of the system with respect to different scales. Moreover, we plan to explore other deep neural network architectures to learn the residual.

REFERENCES

[1] A. B. Zhang and D. Gourley, "Creating digital collections: a practical guide". Oxford: Chandos, 2009.

[2] I. N. Bankman, "Handbook of medical image processing and analysis". Amsterdam: Elsevier - Academic Press, 2009.

[3] L. Yue, H. Shen, J. Li, Q. Yuan, H. Zhang and L. Zhang, "Image super-resolution: The techniques, applications, and future", Signal Processing, vol. 128, pp. 389-408, 2016.

[4] I. Kwang and Y. Kwon, "Single-Image Super-Resolution Using Sparse Regression and Natural Image Prior." IEEE, 2010.

[5] W. Yang, X. Zhang, Y. Tian, W. Wang and J. Xue, "Deep Learning for Single Image Super-Resolution: A Brief Review", 3rd ed. IEEE, 2019.

[6] C. Dong, C. Change Loy and K. He, "Learning a Deep Convolutional Network for Image Super-Resolution". in European Conference on Computer Vision (ECCV 2014).

[7] C. Dong, C. Change Loy, K. He and X. Tang, "Image Super-Resolution Using Deep Convolutional Networks", 3rd ed. IEEE, 2015.

[8] J. Kim, J. K. Lee, and K. M. Lee, "Accurate Image Super-Resolution Using Very Deep Convolutional Networks" 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[9] "Single Image Super-Resolution Using Deep Learning", Mathworks.com. [Online]. Available: https://www.mathworks.com/help /images/single-image-super-resolution-using-deep-learning.html. . [Accessed 25 Sep. 2019].

[10] Glasner, D., Bagon, S., Irani, M.: "Super-resolution from a single image." In: IEEE International Conference on Computer Vision, 2009.

[11] Timofte, R., De Smet, V., Van Gool, L.: "Anchored neighborhood regression for fast example-based super-resolution". In: IEEE International Conference on Computer Vision, 2013.

[12] W. Shi, J. Caballero, F. Huszar, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network", 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[13] C. Du, H. Zewei, S. Anshun, Y. Jiangxin, C. Yanlong, C. Yanpeng, T. Siliang, and M. Y. Yang, "Orientation-Aware Deep Neural Network for Real Image Super-Resolution", 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2019.

[14] J. Yamanaka, S. Kuwashima, and T. Kurita, "Fast and Accurate Image Super Resolution by Deep CNN with Skip Connection and Network in Network", Neural Information Processing Lecture Notes in Computer Science, pp. 217–225, 2017.

[15] Y. Tai, J. Yang, and X. Liu, "Image Super-Resolution via Deep Recursive Residual Network", 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[16] X. Du, X. Qu, Y. He, and D. Guo, "Single Image Super-Resolution Based on Multi-Scale Competitive Convolutional Neural Network", Sensors, vol. 18, no. 3, p. 789, Jun. 2018.

[17] C.-H. Lee, K. Zhang, H.-C. Lee, C.-W. Cheng, and W. Hsu, "Attribute Augmented Convolutional Neural Network for Face Hallucination", 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2018.

[18] E. Zhou, H. Fan, Z. Cao, Y. Jiang and Q. Yin, "Learning face hallucination in the wild", Twenty-Ninth AAAI Conference on Artificial Intelligence, 2015.

[19] A. Watson, "Digital images and human vision". Cambridge, Mass.: MIT Press, 1993.

[20] R. Dosselmann and X. Yang, "A formal assessment of the structural similarity index." Regina: University of Regina, 2008.

[21] S.Pdp. 2019. "Receptive Fields in Convolutional Neural Networks". [online] Medium. Available: https://medium.com/@santi.pdp/receptive-fields-in-convolutional-neural-networks-6368a699d838 [Accessed 16 Nov. 2019].

[22] S. W. Kelsey, "Painting of loom," Notes Queries, vol. s10-III, no. 69, p. 308, 1905.

[23] Large-scale CelebFaces Attributes (CelebA) Dataset. [Online]. Available: http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html.

[24] Z. Cheng, X. Zhu and S. Gong, "QMUL-SurvFace", London: Queen Mary University of London, 2018. [online]. Available: https://qmulsurvface.github.io/.

[25] Image-net.org. "ImageNet". [online] Available: http://www.image-net.org/ [Accessed 23 Oct. 2019].

[26] Vision.ee.ethz.ch. "NTIRE2019: New Trends in Image Restoration and Enhancement workshop and challenges on image and video restoration

and enhancement". [online] Available: http://www.vision.ee.ethz.ch /ntire19/ [Accessed 22 Oct. 2019].

[27] D. Martin and C. Fowlkes and D. Tal and J. Malik, "The Berkeley Segmentation Dataset and Benchmark", California: Berkeley university, 2001. [online]. Available: https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/BSDS300/html/dataset/images.html [Accessed 22 Oct. 2019].

[28] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image Super-Resolution Via Sparse Representation", IEEE Transactions on Image Processing, vol. 19, no. 11, pp. 2861–2873, 2010.

[29] J. Yang, J. Wright, T. Huang, and Y. Ma, "Image super-resolution as sparse representation of raw image patches", 2008 IEEE Conference on Computer Vision and Pattern Recognition, 2008.

[30] A. Nehemiah, "Code for Face Recognition with MATLAB Webinar v1.1.0", Mathworks.com, 2020. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/53849-code-for-face-recognition-with-matlab-webinar

[31] W. T. Freeman and M. Roth, "Orientation Histograms for Hand Gesture Recognition." Cambridge: Mitsubishi Electric Research Laboratories, 1994.

[32] Mathworks.com. 2020. "K-Nearest Neighbor Classification - MATLAB." [online] Available at: https://www.mathworks.com/help/stats/classificationknn.html [Accessed 12 Feb 2020].