



Graphical Processing Unit Accelerated Face Resolution Enhancement using Pixels- Homogeneity and Relative-Ratios

Vikram Mutneja¹, Satvir Singh²

¹Ph.D. Research Scholar, I.K. Gujral Punjab Technical University, Kapurthala, Punjab, India; ²Associate Professor, I.K. Gujral Punjab Technical University Main Campus, Kapurthala, Punjab, India.

ABSTRACT

This work presents a GPU (Graphical Processing Unit) accelerated spatial domain oriented face resolution enhancement algorithm based on the homogeneity levels and relative-ratios of the pixels with respect to its surrounding pixels. The algorithm has been developed, implemented as well as tested in the MATLAB environment. MATLAB is slow in processing but at the same time a resourceful environment for the development in the area of image processing owing to its extremely rich set of functions and programmer-friendly integrated development environment. However, to compensate for the speed loss in testing and implementation phase, we have made use of GPU computing i.e. done parallelization of the algorithm on NVIDIA GPU using CUDA (Compute Unified Device Architecture) interface in the MATLAB environment. It is a simple but efficient algorithm in which kernel matrices are created encoding the homogeneity levels and relative-ratios of all pixels in surrounding four quadrants. Kernel matrices are subsequently applied to reconstruct the HR (High-Resolution) version from input LR (Low-Resolution) facial image.

Key Words: Image processing, GPU computing, Face resolution enhancement, Surveillance videos, Spatial domain processing

INTRODUCTION

The processing of the facial images is an essential task in the smart video surveillance systems. The resolution of the facial images from surveillance videos is usually very less owing to the various factors such as hardware constraints, distance between the subjects and camera. Therefore enhancement concerning resolution is a major step for the purpose of overall improvement in the task of face detection and recognition. As the task of video processing demands of fast computational processing, the parallelization of proposed algorithm has been done on the NVIDIA Graphical Processing Unit (GPU) in CUDA (Compute Unified Device Architecture) under MATLAB environment.

The facial image processing systems can be classified based on different attributes, such as the number of input facial images, face angle, quality and source of images etc. Some of the techniques use single input LR (Low-Resolution) image, while some use multiple LR images which are captured gen-

erally from consecutive frames of video. The latter technique reconstructs the HR image from multiple LR images, therefore called as reconstruction based techniques. The other type of techniques which have attracted large amount of research work in past one decade are the learning based techniques. The relationship between single or multiple LR images and corresponding HR image is learned via a machine learning or intelligent technique. Further, the optimization techniques may be applied for tuning of weights to improve the results. Most of the learning based methods are based on the single facial image as input and use the trained system or dictionary to estimate the missing HR details. Reconstruction based models are based on the generalized smoothness priors while the learning based techniques use recognition based priors.

Based upon the region of interest, the human facial image processing systems can be classified in two major categories. The methods which use the positions of facial landmarks such as left eye, right eye, nose tip and other facial regions are called as local methods (Yang et al. (2010), Liu and Yang

Corresponding Author:

Vikram Mutneja, Ph.D. Research Scholar, I.K. Gujral Punjab Technical University, Kapurthala, Punjab (India).
E-mail: vikram.mutneja@gmail.com.

ISSN: 2231-2196 (Print)

ISSN: 0975-5241 (Online)

DOI: 10.7324/IJCRR.2017.9131

Received: 10.05.2017

Revised: 03.06.2017

Accepted: 20.06.2017

(2014)), while another type of technique is based on template matching, called as global methods. Supposedly the local methods require the input facial images of reasonable high quality so that facial landmarks can be extracted, which also depends on the facial image resolution. It is easier to localize the facial landmarks in the high-resolution facial images as compared to the low-resolution facial image, which necessitates the task of face resolution enhancement.

The development techniques for image processing have certain barriers such as the computational load of handling huge amount of graphics data of images which may involve computation on pixels or their extracted features. At the same time, support from the development environment largely affects the development time. MATLAB is an ideal environment for doing research in the area of image processing because of rich set of functions available in its Image Processing, Computer Vision and related toolboxes. However, MATLAB suffers severely in computational performance as applications written in MATLAB do not run directly on CPU or operating system but the MATLAB engine. On the other hand, developing CUDA (Compute Unified Device Architecture) kernels to harness the power of GPU directly into C or FORTRAN requires writing a lot of glue code for tasks such as communicating data between GPU and CPU, managing GPU memory, initializing and launching CUDA kernels and visualizing their output, which makes it difficult to evaluate and test CUDA kernels for whole of input data space and visualizing their corresponding output. To overcome this limitation, MATLAB's parallel processing toolbox comes to rescue as its GPU computing libraries facilitate to quickly develop, evaluate, visualize and analyze CUDA kernels.

In this work, we have done improvements and GPU acceleration of our previous work on face resolution enhancement Mutneja and Singh (2017). This paper is structured as follows: Section 2 provides Literature Survey. The proposed technique based on pixels-homogeneity and relative-ratios has been explained in Section 3, Section 4 describes experimental setup, Sections 5 & 6 elaborates Results and Discussion respectively, and finally, Sections 7 & 8 gives Conclusions and Future Scope respectively.

Literature Survey

Many researchers have worked in the area of reconstruction based face super-resolution such as by Baker and Kanade (2002), Jiang et al. (2016a), Yang et al. (2010). A lot of work has been done in the learning based face resolution enhancement known as face hallucination such as by Yin et al. (2016), Tu et al. (2017). Some of the works have done hybridization of reconstruction and learning, such as by Yeganli et al. (2016). Many spatial domain based techniques are based on neighbor embedding such as by Chang et al. (2004), Huang et al. (2016), Jiang et al. (2016b).

Some researchers have worked in the field of sparse representation based face super resolution. Yang et al. (2010) presented a single image super-resolution by estimating sparse representation for each patch of the low-resolution input to be applied with the high-resolution image patch dictionary. Zhang et al. (2016) proposed a noise robust method based on information entropy and regression, to weight facial patches for estimating high frequency details from input low-resolution facial images. Huang et al. (2017) Proposed a novel sparse coding-based face hallucination method, by incorporating the intrinsic geometric structure of training samples for dictionary learning. Authors worked to minimize artificial effects using graph construction in HR manifold, and K-selection mean constraints for finding optimal weight HR face reconstruction. Rahiman and George (2017) worked on single image super-resolution using neighbor embedding and sparse representation based learning using partitioned feature space and statistical prediction model. Mao et al. (2016) proposed a new weighted-patch super-resolution method using AdaBoost iteratively to focus more on the patches (face regions) with richer information for improving the reconstruction power.

Dahl et al. (2017) presented a pixel recursive super resolution model using convolutional neural networks to synthesize realistic details into images while enhancing their resolution, represented a multi-modal conditional distribution by accurately modeling the statistical dependencies among the high-resolution image pixels, conditioned on a low-resolution input.

Chen et al. (2017) proposed noisy facial images super-resolution using the contour features to handle noise and standard deviation prior as statistical measure to enhance the low quality contour feature. Jiang et al. (2017) developed facial image super-resolution method based on missing image interpolation information using 'Smooth Regression and a novel Local Structure Prior' (SRLSP). Tu et al. (2017) proposed face hallucination method using direct combined approach, independent of size and number of training samples. Hui et al. (2017) developed appearance, geometrical features, and optical flow model single LR based face resolution enhancement technique. Gong and Wang (2017) developed face hallucination based on similarity measurements between single LR and corresponding multiple HR training images.

Proposed Technique

In this work, GPU acceleration of face resolution enhancement technique has been proposed based on hybridization of homogeneity levels and relative-ratios of pixels in LR patches, by using window based scanning of whole input facial image. Kernel matrices are generated for encoding homogeneity levels as well as relative-ratios to construct resultant HR image. LR patches are slid to the nucleus of HR patches and remaining pixels in HR patch are computed by

application of kernel matrices. The size of LR patch has been selected as 3×3 which is swelled to make the HR patch of size 5×5. For generating the homogeneity kernel matrix, LR window is slid over whole input LR image and checks the homogeneity levels of pixels in the surrounding four quadrants. To check for the homogeneity levels, the summation of pixel intensities is done in the respective quadrant and if found less than the set threshold, is marked as homogeneous otherwise non-homogenous. Figure 1 shows the division of LR patch into four surrounding quadrants. Algorithm 1 shows the working of GPU CUDA Kernel for generating the homogeneity matrix.

```

Algorithm 1 GPU CUDA Kernel for Pixels-Homogeneity Matrix Generation
1: Load the Kernel Constants: NR: Number of Rows, NC: Number of Columns of input image/video frame (SI)
2: Load the Kernel Arguments: Pointer to Input Image (*SI), Pointer to Output Homogeneity Kernel Matrix (*SI_H) to hold the status of Homogeneity levels of all pixels in image, of size (NR × NC)
3: Copy Image to Shared Memory for better efficiency.
4: Load the pixel co-ordinates from thread Indices: X=blockIdx.x, Y=blockIdx.y
5: Compute the linear index of pixel to be operated: linearIdx=X * NR+Y
6: Initialize the Difference Matrix (D) to hold the difference of pixels in the scanning window (3×3) with respect to the central pixel
7: Initialize HCODE=0 to hold the homogeneity levels of pixel, Quadrant wise at Lower Nibble (Q4Q3Q2Q1)
8: Compute the values of Difference Matrix D (3×3), considering SI(ri,ci) as central pixel
9: Compute the sum of values in all four quadrants, stored as H1, H2, H3, H4 from difference matrix (D)
10: Loop1: For i=1:4
11:   if Hi <= Th limit then Set the corresponding bit in HCODE: HCODE = bitor(HCODE; i)
12: Iterate Loop1 for all Pixels in Quadrant
13: Assign HCODE to respective location in output Kernel Matrix SI_H, SI_H[linearIdx] = HCODE
16: Return
    
```

To generate the kernel matrix for the relative-ratios, the ratio of intensity levels of all the pixel is computed with respect to the central pixel in LR patch. The process of iterating the pixels is same as depicted in the Algorithm 1. After generation of kernel matrices, they are applied to the input image to create the corresponding HR image. Algorithm 2 shows the flow chart of proposed technique to generate the HR image by applying the homogeneity and relative-ratios kernel matrices. The GPU kernel thread block size has been set as 1×1×1 while the grid size has been set as *NR×NC×1*.

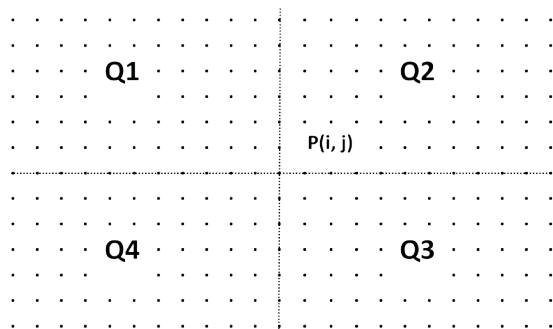


Figure 1: Division of Pixels in Four Surrounding Quadrants for Computing Homogeneity Matrix.

Experimental Setup

The machine running Windows 7 (64 Bit) on Intel Core i3 1.9 GHz and having NVIDIA graphics processing unit GeForce GT 740M has been used to develop and test the proposed method.

RESULTS

Figure 2 shows the results of application of proposed algorithm (shown in column 4) on some of the test LR images (shown in first column) along with the existing techniques ('Bicubic' and 'Box').

Table 1: Results of Testing on Grayscale Images

Quality Metric	Output HR Image Size(450×405) SF=1.67	Output HR Image Size(450×405) SF=2.78
SSIM	0.923	0.869
PSNR	35.679	29.564
SNR	28.196	22.772
MSE	19.852	70.172
CPU Time	0.38 s	0.62 s
GPU Time	0.12 s	0.31 s

Table 1 and Table 2 shows the results of application of proposed algorithm on the Gray-scale and Color test images respectively, for different values of scaling factors.

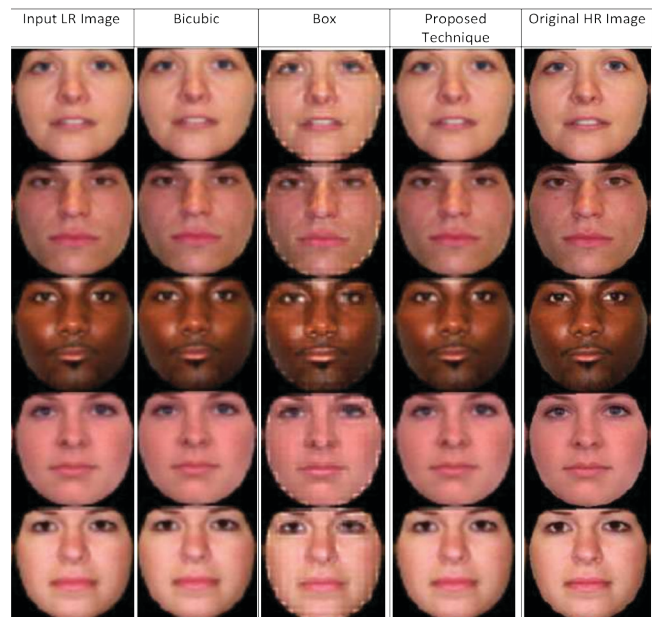


Figure 2: Comparison of the Proposed Technique on Some Test Images.

Table 2: Results of Testing on Color (RGB) Images

Quality Metric	Output HR Image Size(450×405) SF=1.67	Output HR Image Size(450×405) SF=2.78
SSIM	0.972	0.931
PSNR	35.441	29.845
SNR	28.489	23.12
MSE	22.822	76.618
CPU Time	0.74 s	0.93 s
GPU Time	0.53 s	0.67 s

Further the gain in the time efficiency achieved by applying the parallelization technique through GPU computing as compared to its serial version is depicted in Figure 3.

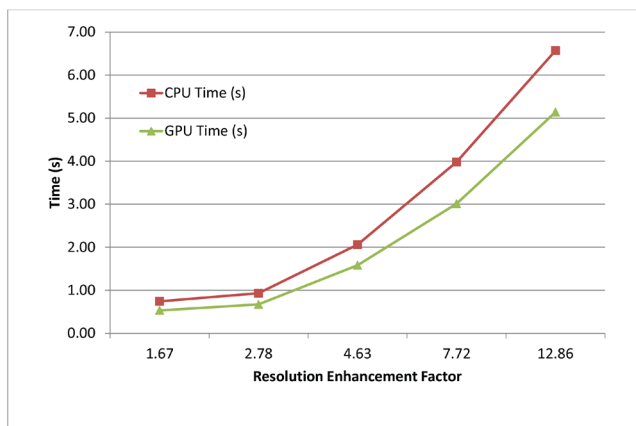


Figure 3: Comparison of Time Efficiency CPU (Serial Version) Versus GPU (Parallel Version)

DISCUSSION

The quality metrics Structural Similarity Index Measure (*SSIM*), Peak Signal To Noise Ratio (*PSNR*), Signal To Noise Ratio (*SNR*) and Mean Square Error (*MSE*) have been measured from the generated HR image with respect to the original HR image, along with the measurement of GPU acceleration achieved with respect to the CPU (Central Processing Unit) code (refer Table 1 & 2). As the work has been done in MATLAB, the comparison of results of proposed algorithm has been done with some methods ('Bicubic' and 'Box') available in MATLAB (refer Figure 3).

The proposed method has been tested on color as well as gray-scale images. Two folders have been created for testing, each containing 20 test images. One of the folders contains gray-scale images and second constitutes color facial images from database color FERET (Phillips et al. (2000, 1998)). Portions of the research in this paper use the FERET database of facial images collected under the FERET program, sponsored by the DOD Counterdrug Technology Development Program Office FERET Phillips et al. (1998), Phillips

et al. (2000). Table 1 and Table 2 shows the mean values of measured quality metrics and time taken for gray-scale and color images respectively. The term SF refers to the scaling factor i.e. face resolution enhancement factor. By comparison, it is inferred that application of proposed technique using color based processing (i.e. of channels R,G and B) gives better results as compared to gray-scale based only processing on color images.

Further, the efficiency of the proposed algorithm has been tested on different resolution factors. For this, the test images from FERET are down-scaled to the factors 0.6000, 0.3600, 0.2160, 0.1296 and 0.0778. The percentage reduction in the time cost has been calculated in each case. Figure 3 shows the comparison of time cost of proposed algorithm using GPU Computing with respect to the CPU code. The maximum percentage time cost reduction of the order of 28.38% has been achieved corresponding to the scaling factor 1.67 and 22.39% at scaling factor of 12.86.

CONCLUSION

We have successfully developed and tested the GPU accelerated face resolution enhancement algorithm based on pixels-homogeneity and relative-ratios in MATLAB environment using CUDA based parallel computing. The testing of the proposed algorithm has been done on test facial images from web and FERET database (Phillips et al. (2000, 1998)) and comparison done with some of the existing techniques in terms of quality metrics *SSIM*, *PSNR*, *SNR* and *MSE*. It has been observed that considerable speedup was achieved by the parallel execution of proposed face resolution enhancement algorithm as compared to its serial version.

As MATLAB interprets the code, and it is the interpreter that slows down the processing, therefore by using GPU computing, we have tried to compensate for this time efficiency loss and at the same time benefitting user friendly environment of MATLAB for efficient development of GPU computing based algorithm. From the results of quality metrics and time efficiency improvement achieved by parallelization, we conclude that proposed system is very effective and efficient in resolution enhancement of facial images.

Future Scope

In the near future, we intend to make improvements in the proposed technique by making use of shared memory in GPU, increasing the extent of parallelization for further time acceleration, introducing adaptability in terms of selection of size of LR and HR patches for achieving the desired resolution enhancement factors. Further, we intend to hybridize the proposed technique with boosting based analysis for improvement in the quality of resultant HR facial image.

Algorithm 2 Resolution Enhancement Using Pixels-Homogeneity Relative-Ratios Kernels

```

1: Load the Input Image/Frame from Video, SI
2: Get the Size of the Input LR (Low Resolution) Image, SNR=Number of Rows, SNC: Number of Columns
3: Get the Gray-Scale version or Respective channel (R,G or B) of Input Image (SIC)
4: Initialize GPU CUDA Kernels: Kernel1 for Homogeneity Matrix and Kernel2 for Relative-Ratios Matrix
5: Initialize GPU CUDA Kernel Constants: NR (Number of Rows), NC (Number of Columns) and Kernel dimensions
6: Initialize GPU CUDA Kernel Arguments: SI (Input Image), SI_H (Homogeneity Kernel Matrix) and SI_RR (Relative Ratios Kernel Matrix)
7: Get the Homogeneity Kernel Matrix by invoking CUDA Kernel1, Kernel Arguments: (SI, SI_H)
8: Get the Relative Ratios Kernel Matrix by invoking CUDA Kernel2, Kernel Arguments: (SI, SI_RR)
9: Initialize the matrix TIC to hold the target HR image of size, Number of rows  $TNR = (5/3) * SNR$ , Number of columns  $TNC = (5/3) * SNC$ , Class: Unsigned int 8 bit.
10: Initialize the Temporary Matrix T( $5 \times 5$ ) as envelope to hold  $3 \times 3$  patch from LR image, and Difference Matrix D( $3, 3$ ) to hold the difference of neighborhood pixels w.r.t. central pixel
11: Loop1: For  $ri = 2 : 3 : SNR - 1$  (Nested Loops to Scan the pixels in LR Image)
12:   Loop2: For  $ci = 2 : 3 : SNC - 1$ 
13:     Insert the ( $3 \times 3$ ) patch from LR image to center of T i.e. to rows and columns 2 to 4 of T
14:     Compute the values of Difference Matrix D ( $3, 3$ ), considering S(ri,ci) as central pixel
15:     Find mean of pixel values in all four quadrants as A1, A2, A3 and A4, in  $3 \times 3$  LR patch
16:     Get the homogeneity code for pixel location  $r_i, c_i$ , Check the status of homogeneity of All four quadrants in LR patch
17:     Loop3: For  $i = 1 : 4$  To Scan for Pixels in a Quadrant
18:       if  $H_i = 1$  then
19:         Homogeneous Quadrant, Assign to the unassigned pixels of envelope T the corresponding mean quadrant value (Ai),  $i = 1$  to 4
20:       else
21:         Non Homogeneous Quadrant, Translate the difference matrix D of LR patch to respective corner ( $3 \times 3$  patch) in envelope T (HR Patch of size  $5 \times 5$ )
22:       Iterate Loop3, For All Pixel Locations in Quadrant
23:       Multiply the generated ( $5 \times 5$ ) patch with respective patch from Relative-Ratios kernel to generated resultant HR patch
24:       Compute the target locations for inserting the computed  $5 \times 5$  HR patch in the resultant HR image (TIC)
25:     Iterate Loop2 for next pixel location
26:   Iterate Loop1 for next pixel location
27: HR Image TIC Generated

```

ACKNOWLEDGEMENT

Authors gratefully acknowledge the support provided by I.K.G. Punjab Technical University, Kapurthala, Punjab, India. Authors deeply acknowledge the immense help received from the scholars whose articles are cited and included in references of this manuscript. The authors are also grateful to authors / editors / publishers of all those articles, journals and books from where the literature for this article has been reviewed and discussed.

REFERENCES

- Baker, S. and Kanade, T. (2002). Limits on super-resolution and how to break them. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(9):1167–1183.
- Chang, H., Yeung, D.-Y., and Xiong, Y. (2004). Super-resolution through neighbor embedding. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 1, pages I–I. IEEE.
- Chen, L., Hu, R., Liang, C., Li, Q., and Han, Z. (2017). A novel face super resolution approach for noisy images using contour feature and standard deviation prior. *Multimedia Tools and Applications*, 76(2):2467–2493.
- Dahl, R., Norouzi, M., and Shlens, J. (2017). Pixel recursive super resolution. *arXiv preprint arXiv:1702.00783*.
- Gong, M. and Wang, K. (2017). Single face hallucination via local neighbor patches. *AEU-International Journal of Electronics and Communications*, 74:88–93.
- Huang, K., Hu, R., Jiang, J., Han, Z., and Wang, F. (2016). Face image super-resolution through improved neighbor embedding. In *International Conference on Multimedia Modeling*, pages 409–420. Springer.
- Huang, K., Hu, R., Jiang, J., Han, Z., and Wang, F. (2017). Hrm graph constrained dictionary learning for face image super-resolution. *Multimedia Tools and Applications*, 76(2):3139–3162.
- Hui, Z., Liu, W., and Lam, K.-M. (2017). A novel correspondence-based face-hallucination method. *Image and Vision Computing*.
- Jiang, J., Chen, C., Huang, K., Cai, Z., and Hu, R. (2016a). Noise robust position-patch based face super-resolution via tikhonov regularized neighbor representation. *Information Sciences*, 367:354–372.
- Jiang, J., Chen, C., Ma, J., Wang, Z., Wang, Z., and Hu, R. (2017). Srrsp: A face image super-resolution algorithm using smooth regression with local structure prior. *IEEE Transactions on Multimedia*, 19(1):27–40.
- Jiang, J., Hu, R., Wang, Z., Han, Z., and Ma, J. (2016b). Facial image hallucination through coupled-layer neighbor embedding. *IEEE Transactions on Circuits and Systems for Video Technology*, 26(9):1674–1684.
- Liu, S. and Yang, M.-H. (2014). Compressed face hallucination. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 4032–4036. IEEE.
- Mao, S., Zhou, D., Zhang, Y., and Zhang, Z. (2016). A boosting method to face image super-resolution. *arXiv preprint arXiv:1609.01805*.
- Mutneja, V. and Singh, S. (2017). A novel facial image resolution enhancement algorithm using pixels homogeneity. *International Journal*, 8(3).
- Phillips, P. J., Moon, H., Rizvi, S. A., and Rauss, P. J. (2000). The feret evaluation methodology for face-recognition algorithms. *IEEE Transactions on pattern analysis and machine intelligence*, 22(10):1090–1104.
- Phillips, P. J., Wechsler, H., Huang, J., and Rauss, P. J. (1998). The feret database and evaluation procedure for face-recognition algorithms. *Image and vision computing*, 16(5):295–306.
- Rahiman, V. A. and George, S. N. (2017). Single image super resolution using neighbor embedding and statistical prediction model. *Computers & Electrical Engineering*.
- Tu, C.-T., Ho, M.-C., and Lin, M.-Y. (2017). A new approach for face hallucination based on a two-dimensional direct combined model. *Pattern Recognition*, 62:1–20.
- Yang, J., Wright, J., Huang, T. S., and Ma, Y. (2010). Image super-resolution via sparse representation. *IEEE transactions on image processing*, 19(11):2861–2873.
- Yeganli, F., Nazzal, M., Unal, M., and Ozkaramanli, H. (2016). Image super-resolution via sparse representation over multiple learned dictionaries based on edge sharpness. *Signal, Image and Video Processing*, 10(3):535–542.
- Yin, Q., Cao, Z., and Zhou, E. (2016). Face hallucination using convolutional neural networks. *US Patent 9,405,960*.
- Zhang, Y., Zhang, Z., Hu, G., and Hancock, E. R. (2016). Face image super-resolution via weighted patches regression. In *Pattern Recognition (ICPR), 2016 23rd International Conference on*, pages 3892–3897. IEEE.