Continental Project under ADAS

Collaborative Research Project with IIT, Delhi
PI: Dr.Subhashis Banerjee, IIT Delhi.

Project Title:

Restoration of Motion Blurs and Image Super Resolution

Project Report

2016-17

By

Dr.Uma Mudenagudi

uma@bvb.edu

Department of Electronics and Communication,
B. V. Bhoomaraddi College of Engineering and Technology,
Vidyanagar, Hubballi-580031,
Karnataka

Contents

1. Introduction 4
2. Image Super Resolution 6
   - 2.1. Methods tried for super resolution 6
     - 2.1.1. Steering Kernel based Super Resolution 6
     - 2.1.2. Example based Super Resolution 7
     - 2.1.3. Clustering and Collaborative Representation for Fast Single Image Super Resolution (CCR) 8
     - 2.1.4. Learning a Deep Convolution Network for Image Super Resolution (SRCNN) 8
     - 2.1.5. Proposed method for super resolution 9
       - Multi-stage SRCNN for Image Super Resolution 9
   - 2.2. Vision Tools 10
     - 2.2.1. Optical Character Recognition 10
     - 2.2.2. Google Cloud Vision API 10
   - 2.3. Results and Discussions 11
3. Restoration of Motion Blur 19
   - 3.1. Methods tried for restoration of motion blur 19
     - 3.1.1. Steering Kernel 19
     - 3.1.2. Cepstral Method 19
     - 3.1.3 Proposed method for restoration of motion blur 21
       - Nearest Neighbour framework for estimation of motion blur. 21
   - 3.2 Results and Discussions 22
4 Contributions and Future directions 28
   - 4.1 Contributions 28
   - 4.2 Future directions 28

References 29

Abstract

In this project, we address the problem of restoration of motion blur and image super resolution. To address the problem of super resolution we propose Multi-stage SRCNN (convolution neural network) approach. The goal of super resolution is to obtain more informative and/or high resolution image. To address the problem of restoration of motion blur, we propose nearest neighbour framework to determine the PSF by estimating the length and angle of motion blur in the image. We de-convolve the blurred image with the determined PSF to obtain de-blurred image. Generally, blurred image is modelled as a convolution between the original image and the unknown point spread function (PSF). Motion blur occurs when there is relative motion between the camera and the object being captured. We perform restoration of motion blur and image super resolution keeping ADAS (Advanced Driver Assistive Systems) applications into consideration. We demonstrate the quality of the de-blurred and super resolved image using different vision tools.

1. Introduction

In this project, we address the problem of restoration of motion blur and image super resolution. We perform restoration of motion blur and image super resolution keeping ADAS (Advanced Driver Assistive Systems) applications into consideration. Image super-resolution (SR) is a classical problem in computer vision. Many Methods are proposed in literature on single and multi image super resolution [][][][][][]. Recent state-of-the-art methods for single image super-resolution are mostly example-based. These methods either exploit internal similarities of the same image, or learn mapping functions from external low and high-resolution exemplar pairs. The external example-based methods are often provided with abundant samples, but are challenged by the difficulties of effectively and compactly modelling the data.

There are many multi image super resolution methods are also proposed in the literature which perform well in controlled conditions.

In this project, we try working with some multi and some single image super resolution methods which suit the ADAS application. We see the SRCNN has several appealing properties. First, its structure is intentionally designed with simplicity in mind, and yet provides superior accuracy comparing with state-of-the-art example-based methods. Second, with moderate numbers of filters and layers, this method achieves fast speed for practical on-line usage even on a CPU. SRCNN is faster than a series of example-based methods, because it is fully feed-forward and does not need to solve any optimization problem on usage. Third, experiments show that the restoration quality of the network can be further improved when (i) larger datasets are available, and/or (ii) a larger model is used. This motivates us to use the SRCNN framework iteratively to perform super resolution, and we name this as multi-stage SRCNN. In the proposed multi-stage SRCNN, we perform super resolution using 2x training model iteratively (in stagess) to achieve higher magnification.

We also address the problem of restoration of motion blur. The blurred image is modelled as a convolution between the original image and an unknown point-spread function. The angle of motion blur is estimated using three different approaches. Motion blur occurs when there is relative motion between the camera and the object being captured. In this report we study motion blur, that is, blur that occurs when the motion has constant speed and a fixed direction. The goal is to identify the angle and length of the blur. Once the angle and length of the blur are determined, a point spread function can be constructed. This point spread function is then used in direct deconvolution methods to help restore the degraded image. The process of blurring can be modelled as the following convolution,

$$g(x, y) = f(x, y) * h(x, y) + n(x, y), \qquad (1)$$

where, $f(x, y)$ is the original image, $h(x, y)$ is the blurring point spread function, $n(x, y)$ is white noise and $g(x, y)$ is the degraded image.

Many blur removal and restoration algorithms are proposed in the literature. Most of these fail to estimate the length and angle of motion blur introduced. To address this, we propose to estimate the length and angle of motion blur by modelling a nearest neighbour framework.

Towards this;

- We propose a Multi-stage SRCNN method to super resolve the image.
- We propose to estimate the motion blur in an image by modelling a nearest neighbour framework.
- We propose to demonstrate the quality of super resolved image using different vision tools.
- We demonstrate the results on real and synthetic dataset.

We discuss the state of art techniques of image super resolution algorithms and propose multi-stage SRCNN method for super resolution in Section 2. In Section 3, we discuss different methods of motion blur estimation and propose a nearest neighbour framework to estimate the motion blur. We conclude in Section 4.

2. Image Super Resolution

In this section, we discuss the state of art techniques of image super resolution algorithms and propose multi-stage SRCNN method for super resolution. We discuss both multi and single image super resolution algorithms keeping ADAS applications into consideration. We compare the results of different methods with our proposed algorithm using different vision tools.

2.1 Methods Tried for Super resolution

In this section we discuss about different methods that suit the ADAS applications. We discuss both multi image and single image super resolution algorithms form literature and are as listed below:

- Steering Kernel based multi image super resolution.
- Example based single image super resolution.
- Clustering and Collaborative Representation for fast single image super resolution (CCR).
- Learning a Deep Convolution Network for Image Super Resolution (SRCNN).

2.1.1 Steering Kernel based Super Resolution [1]

Steering Kernel based Super Resolution uses two step approach to super resolve an image. This method is a multi image super resolution algorithm. In the first step, an initial estimate of the image gradients is made using second order classical kernel regression. This estimate is used to measure the dominant orientation of the local gradients in the image. This is the first stage of filtering. In the second stage, the orientation information is used to adaptively steer the local kernel, resulting in elongated, elliptical contours spread along the directions of the local edge structure. With these locally adapted kernels, the denoising is affected most strongly along the edges, rather than across them, resulting in strong preservation of details in the super resolved image.

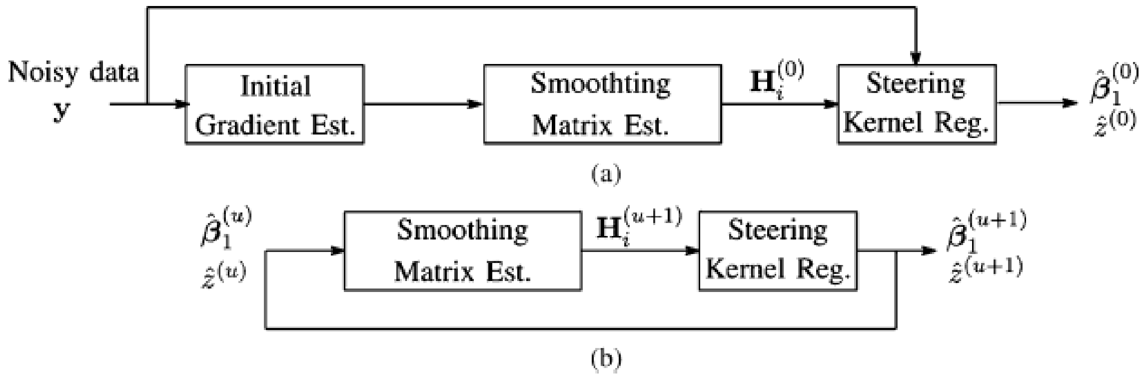


Figure 1: Steering kernel based image super resolution

In the Figure 1, 'u' is the iteration number. The data samples are used to create the initial estimate of the interpolated output image in Figure 1a. In the next iteration, the reconstructed image is used to calculate a more reliable estimate of the gradient figure 1b and this process continues for a few more iterations.

2.1.2 Example based Super Resolution [2]

Example based Super Resolution method learns the mapping between the HR and LR image using Kernel Ridge Regression (KRR). The training images for the regressor are obtained by blurring and sub sampling a set of high resolution images to constitute a set of low and high resolution image pairs. To increase the efficiency of the training set, the data is contrast normalized. During the construction of the training set both the input patch and corresponding desired patch are normalized using L1 norm of the input patch. For an unseen patch, the input is again normalized before the regression and the corresponding output is inverse normalized. KRR tend to fit the data with a smooth function. Hence, to preserve the major edges, the author uses Natural Image Prior (NIP) which focuses on high frequency components of the image, giving the knowledge of discontinuity in image. NIP is used only on the major edges of the image. Major edges are found by thresholding each pixel of the Laplacian of the input image using L2 or L infinity norms of the local patches encompassing it. Finally, this optimized image is combined with bicubic interpolated image to get HR image.

NIP - Natural image prior basically means the edges of the image should be distributed by a hyper laplacian distribution, just like how edges of images of "natural" objects or scenes are distributed. L2 prior (Gaussian prior) don't work well, as they don't in deconvolution, super resolution, or other types of image algorithms.

2.1.3 Clustering and Collaborative Representation for Fast Single Image Super Resolution [3]

In this method two sets of features are extracted, one from HR images and other from LR images from a training image set and cluster them into numerous subspaces. Then, these are used to group neighbour features followed by computing multiple projection matrices. The HR features chosen are high frequency components of HR image and are obtained by subtracting the bicubic interpolated LR image from the corresponding HR image. For LR image, features are first order and second order gradients in horizontal and vertical directions. To reduce complexity, the LR feature space is reduced by applying PCA to the feature set. The LR and HR feature space is clustered using K-Means clustering algorithm. In order to preserve local geometry and group the local manifold around the LR cluster centers, the author uses K-Means to split the LR features spaces into certain number of subspaces and cluster centers. Next, nearest LR and HR feature subspace are grouped. Collaborative representation with L2 norm least square regression is used to obtain projection matrix to map LR feature space to HR feature space. For an input image, the features are clustered as done in training, then using the projection matrix HR patch is obtained.

2.1.4 Learning a Deep Convolution Network for Image Super Resolution (SRCNN) [4]

The super resolution using CNN is performed in three following steps:

Patch extraction – This operation extracts patches from LR image and represents each patch as a high dimensional vector. Patches are convolved with certain filters like Gaussian filters, edge detectors and texture detectors etc.

Non-linear mapping – This operation nonlinearly map each high dimensional vector onto another high dimensional vector. Each mapped high dimensional vector is a high-resolution patch. Nonlinear mapping is applying of another ‘n’ number of filters on the convolved patches.

Reconstruction – This operation aggregates the high-resolution patch to generate image. The reconstruction operation is similar to applying another filter on the feature maps. Hence, the CNN model discussed is of three convolutional layer model.

2.2 Proposed Multi-stage SRCNN for Single image super resolution

In multi-stage SRCNN method, we perform super resolution using 2x training model iteratively (in stages) to achieve higher magnification. Here, we use the same approach of training as in SRCNN method. In this method, to super resolve an image to 4x, we first super resolve the image to 2x and then super resolving this image to 2x to get a final 4x super resolved image as shown in Figure 2.

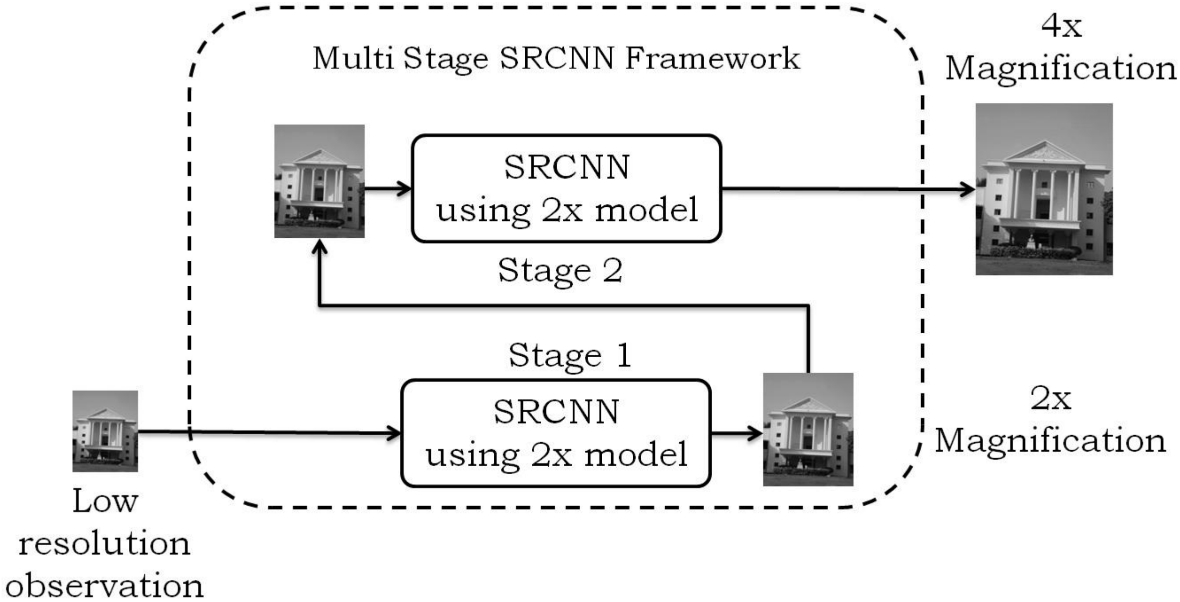


Figure 2: Proposed multi-stage SRCNN Framework

2.3 Vision Tools

In this section, we discuss about the vision tools that we use to demonstrate the results of image super resolution and restoration of motion blur.

2.3.1 Optical Character Recognition

The results show how the OCR detects regions in an image that contain text. This is a common task performed on unstructured scenes. Unstructured scenes are images that contain undetermined or random scenarios [9]. For example, you can detect and recognize text automatically from captured video to alert a driver about a road sign. This is different than structured scenes, which contain known scenarios where the position of text is known beforehand. Segmenting text from an unstructured scene greatly helps with additional tasks such as optical character recognition (OCR) [7]. The automated text detection algorithm in this example detects a large number of text region candidates and progressively removes those less likely to contain text [8]. Following are the steps used to detect text in image:

Step 1: Detect Candidate Text Regions Using MSER

The MSER feature detector works well for finding text regions [6]. It works well for text because the consistent color and high contrast of text leads to stable intensity profiles.

Step 2: Remove Non-Text Regions Based On Basic Geometric Properties

Step 3: Remove Non-Text Regions Based On Stroke Width Variation

Step 4: Merge Text Regions For Final Detection Result

Step 5: Recognize Detected Text Using OCR

2.3.2 Google Cloud Vision API

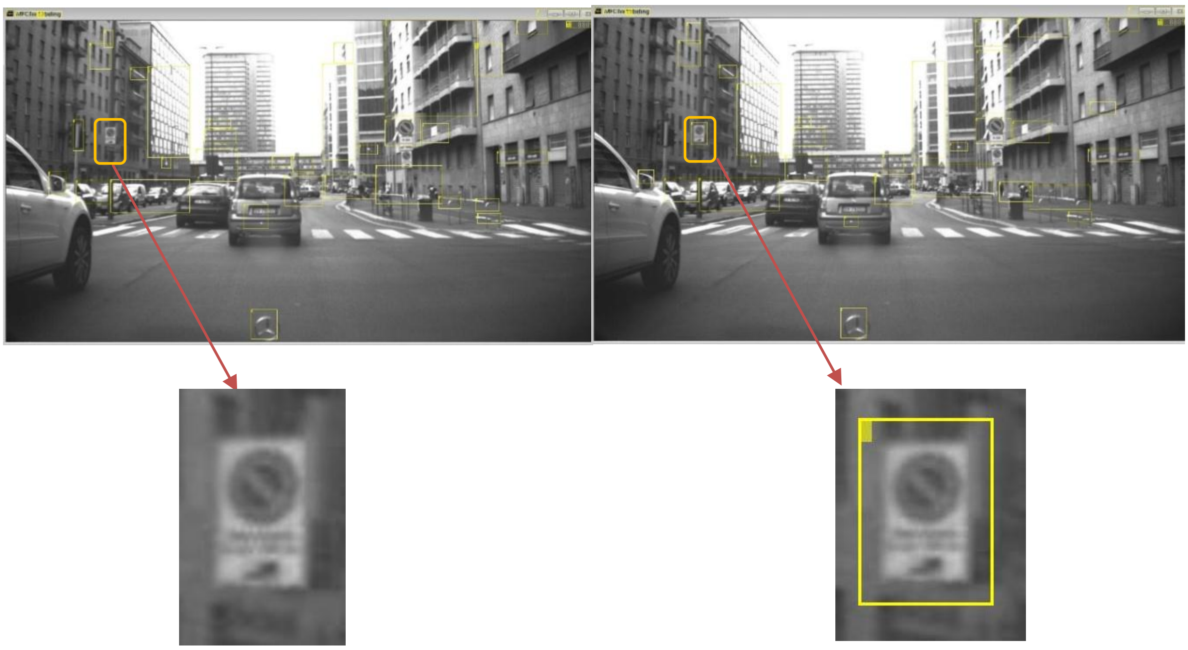
Google Cloud Vision API enables developers to understand the context of image by encapsulating powerful machine learning models in an easy to use REST API. It quickly classifies images into thousands of categories, detects individual objects and faces within images and finds and reads printed words contained within images. One can build metadata on the image catalogue, moderate offensive content, or enable new marketing scenarios through image sentiment analysis. We use Google vision API to demonstrate the results of super resolution.

2.4 Results and Discussions

In this section, we demonstrate and compare the results of different methods of super resolution with our proposed super resolution algorithm using different vision tools.

2.4.1 Results: Steering Kernel based Super Resolution

Figure 2(a) shows the probable areas containing characters detected in bicubic interpolation of *frame 3* from video *test1* and Figure 2(b) shows corresponding super resolved frame using steering kernel. We obseve, more areas containing signal are detected in case of super resolved frame using steering kernel method.



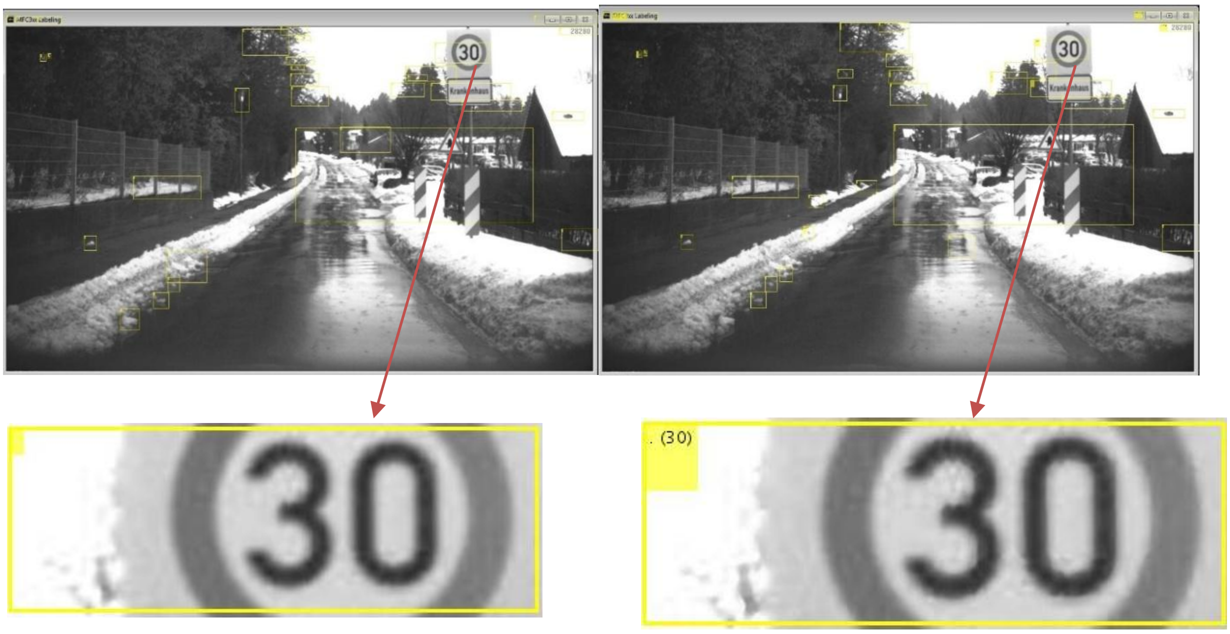
(a) Bicubic interpolation of *frame 3* from video *test1*

(b) Super resolution (4x) of *frame 3* from video *test1* using steering kernel

Figure 2: Comparison of bicubic interpolation with super resolution using steering kernel method

2.4.2 Results: Example based Super Resolution

Figure 3(a) shows the probable areas containing characters detected in bicubic interpolation of *frame 189* from video *2* and Figure 3(b) shows corresponding super resolved frame using example based super resolution. We can see that the number in the signal is detected in the super resolved frame using example based super resolution.



(a) Bicubic interpolation of *frame* 189 from video *2*

(b) Super resolution (4x) of *frame 189* from video *2 using* Example based Super Resolution

Figure 3: Comparison of bicubic interpolation with super resolution using Example based Super Resolution.

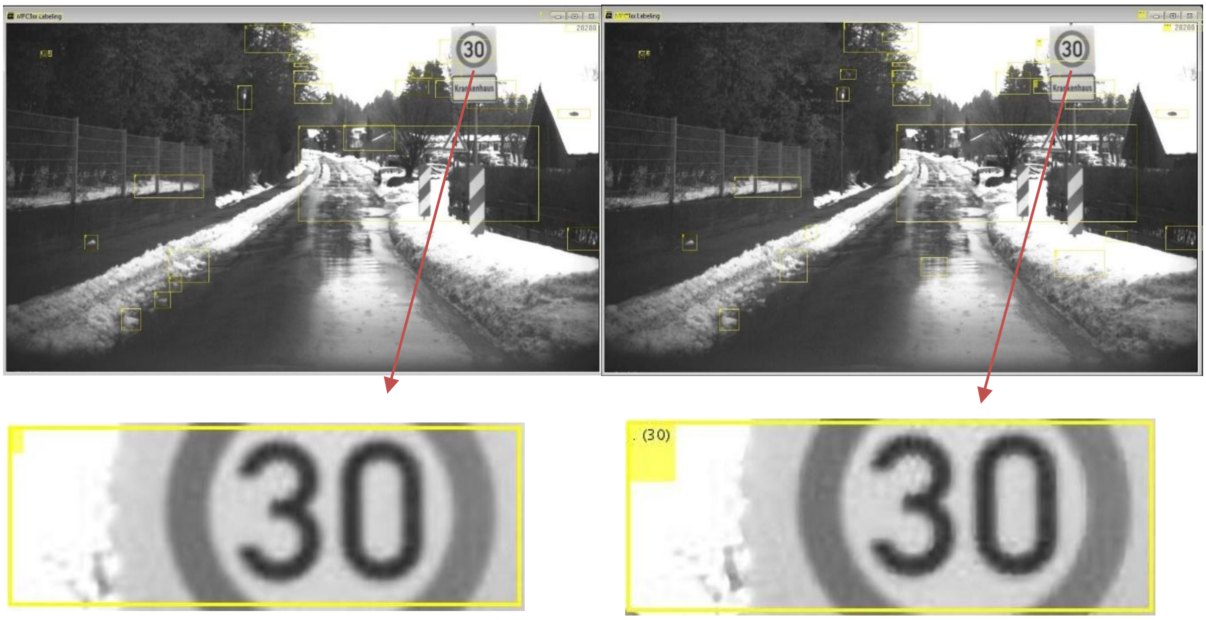
Figure 4(a) shows the maximally stable external regions MSER, which are probable areas containing characters through blobs in bicubic interpolation of *frame 1* from video *1* and Figure 4(b) shows corresponding super resolved frame using Example based Super Resolution. We can see that the more number of blobs/regions detected in the super resolved frame using Example based Super Resolution.

(a) Bicubic interpolation of *frame* 1 from video *1*

(b) Super resolution (4x) of *frame 1* from video *1*

Figure 4: Comparison of MSER in bicubic interpolation with super resolution using Example based Super Resolution.

2.4.3 Results: Clustering and Collaborative Representation for Fast Single Image Super Resolution (CCR).



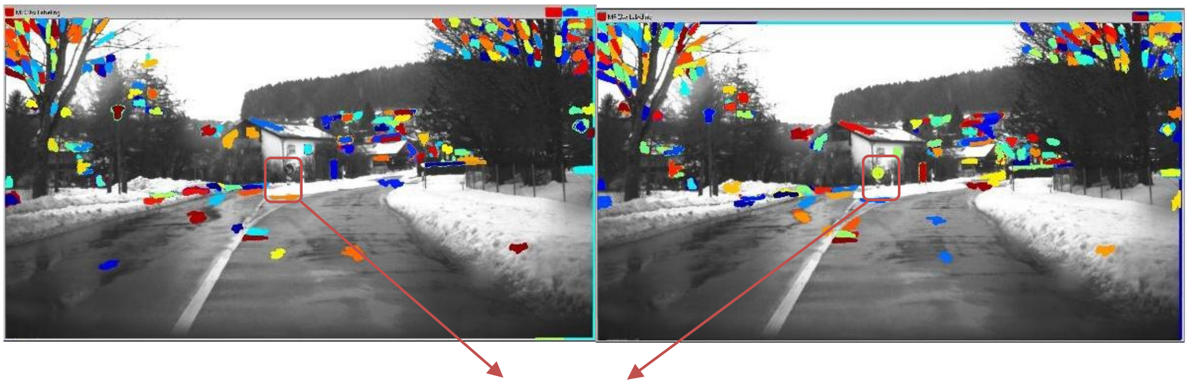
(a) Bicubic interpolation of *frame* 189 from video *2*

(b) Super resolution (4x) of *frame 189* from video *2 using CCR*

Figure 5: Comparison of bicubic interpolation with super resolution using CCR.

Figure 5(a) shows the probable areas containing characters detected in bicubic interpolation of *frame 189* from video *2* and Figure 5(b) shows corresponding super resolved frame using CCR. We can see that the number in the signal is detected in the super resolved frame using CCR.

Figure 6(a) shows the maximally stable external regions MSER, which are probable areas containing characters through blobs in bicubic interpolation of *frame 1* from video *1* and Figure 6(b) shows corresponding super resolved frame using CCR. We can see that the more number of blobs/regions detected in the super resolved frame using CCR method.



In the ROI shown the blob of signal is detected in CCR based SR

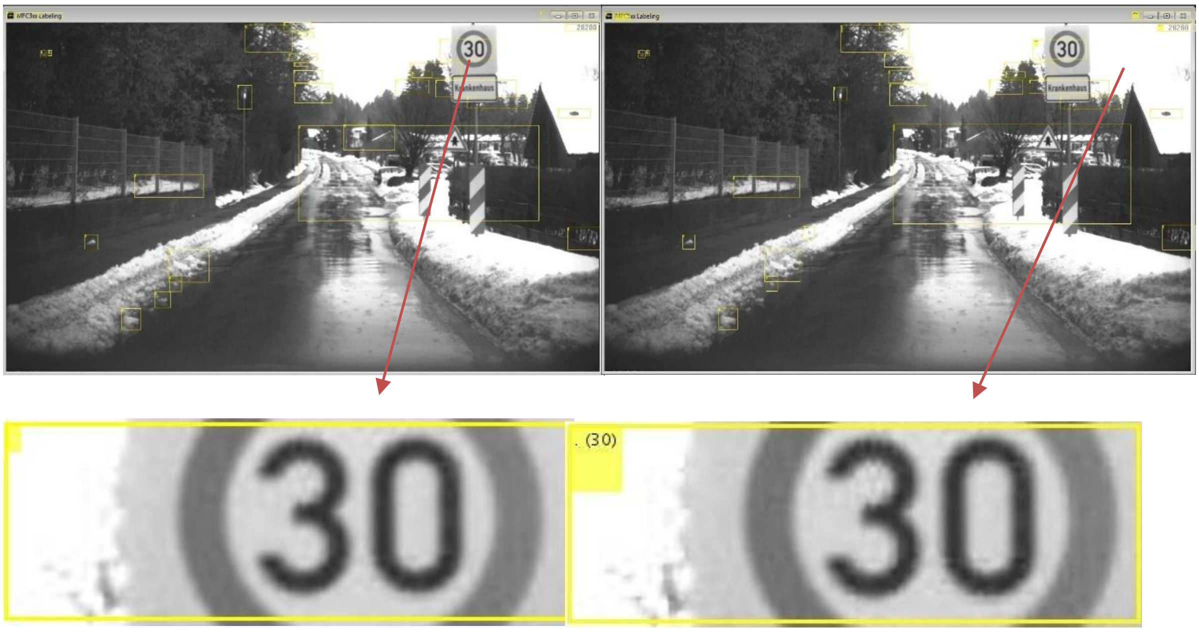
(a) Bicubic interpolation of *frame* 1 from video *1*

(b) Super resolution (4x) of *frame 1* from video *1 using CCR*

Figure 6: Comparison of MSER in bicubic interpolation with super resolution using CCR.

2.4.4 Results: Learning a Deep Convolution Network for Image Super Resolution.

Figure 7(a) shows the probable areas containing characters detected in bicubic interpolation of *frame 189* from video *2* and Figure 7(b) shows corresponding super resolved frame using steering kernel. We can see that the number in the signal is detected in the super resolved frame.

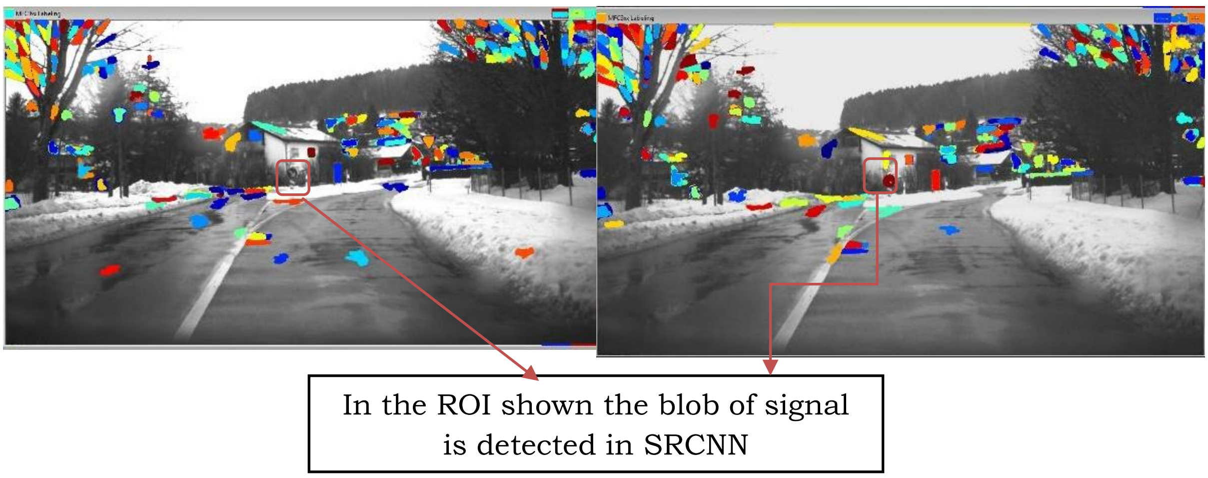


(a) Bicubic interpolation of *frame* 189 from video *2*

(b) Super resolution (4x) of *frame 189* from video *2 using SRCNN*

Figure 7: Comparison of bicubic interpolation with super resolution using SRCNN.

Figure 8(a) shows the maximally stable external regions MSER, which are probable areas containing characters through blobs in bicubic interpolation of *frame 1* from video *1* and Figure 8(b) shows corresponding super resolved frame using SRCNN. We can see that the more number of blobs/regions detected in the super resolved frame using SRCNN method.



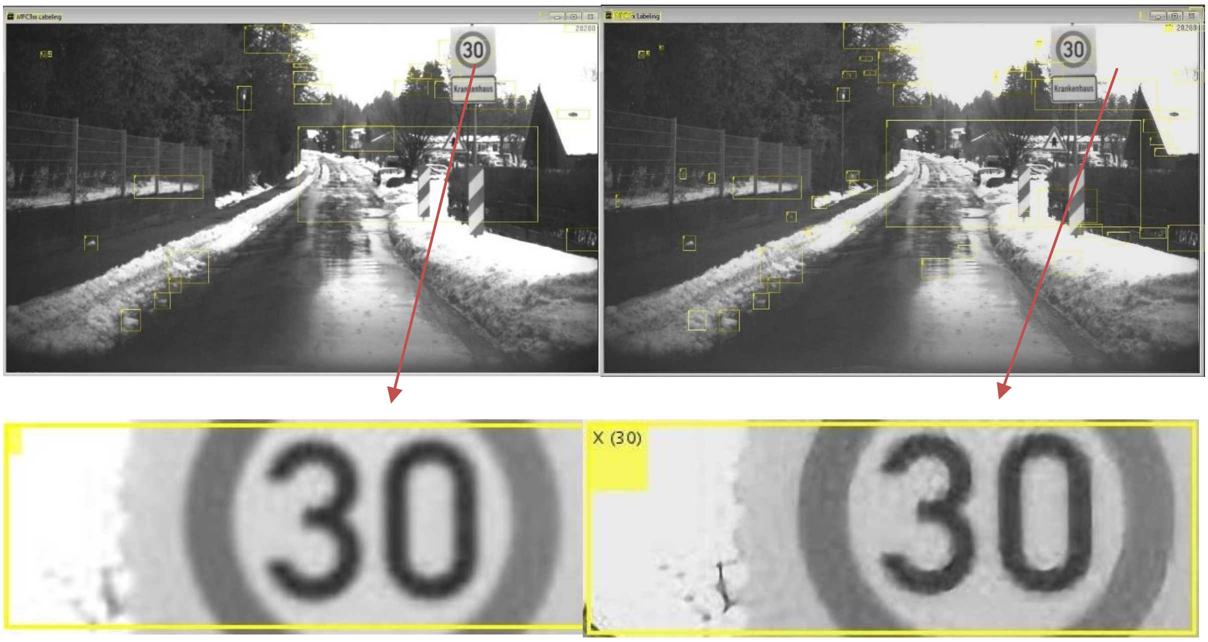
(a) Bicubic interpolation of *frame* 1 from video *1*

(b) Super resolution (4x) of *frame 1* from video *1 using SRCNN*

Figure 8: Comparison of MSER in bicubic interpolation with super resolution using SRCNN.

2.4.5 Results: Proposed multi-stage SRCNN for single image super resolution

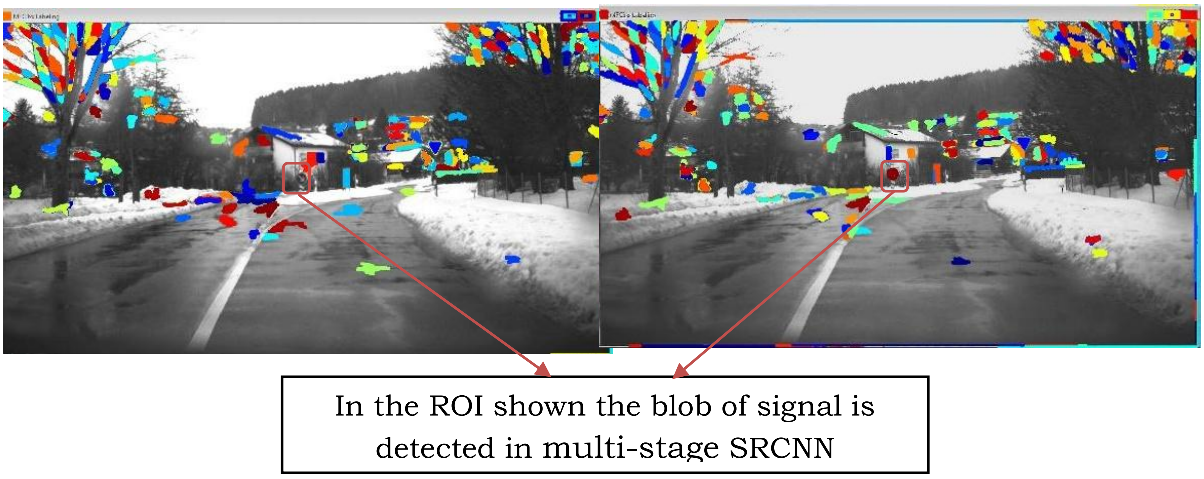
Figure 9(a) shows the probable areas containing characters detected in bicubic interpolation of *frame 189* from video *2* and Figure 9(b) shows corresponding super resolved frame using step SRCNN. We can see that the number in the signal is detected in the super resolved frame.



(a) Bicubic interpolation of *frame* 189 from video *2*

(b) Super resolution (4x) of *frame 189* from video *2 using* multi-stage SRCNN

Figure 9: Comparison of bicubic interpolation with super resolution using multi-stage SRCNN.



(a) Bicubic interpolation of *frame* 1 from video *1*

(b) Super resolution (4x) of *frame 1* from video *1 using* multi-stage *SRCNN*

Figure 10: Comparison of MSER in bicubic interpolation with super resolution using step SRCNN.

Figure 10(a) shows the probable areas containing characters through blobs in bicubic interpolation of *frame 1* from video *1* and Figure 10(b) shows corresponding super resolved frame using Step SRCNN. We can see that the more number of blobs/regions detected in the super resolved frame using SRCNN method.

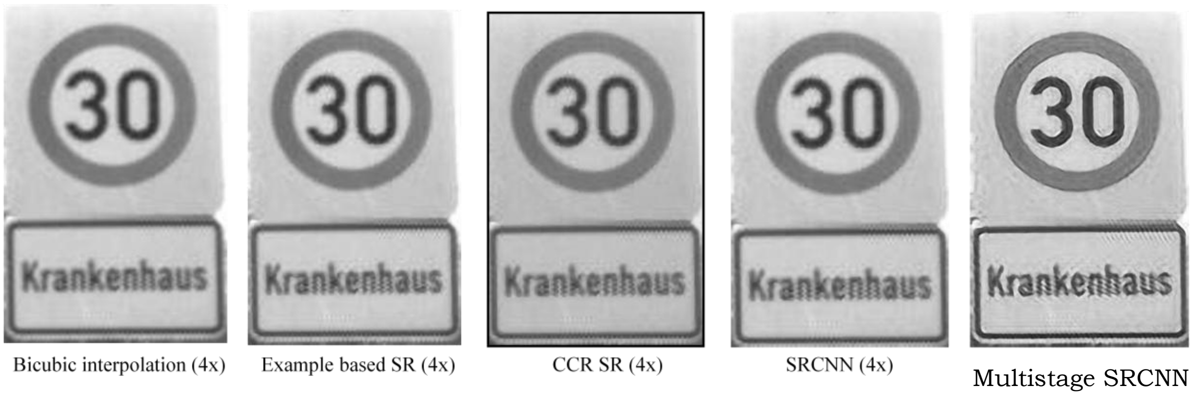


Figure 11: Comparison of proposed multi-stage SRCNN with other methods.

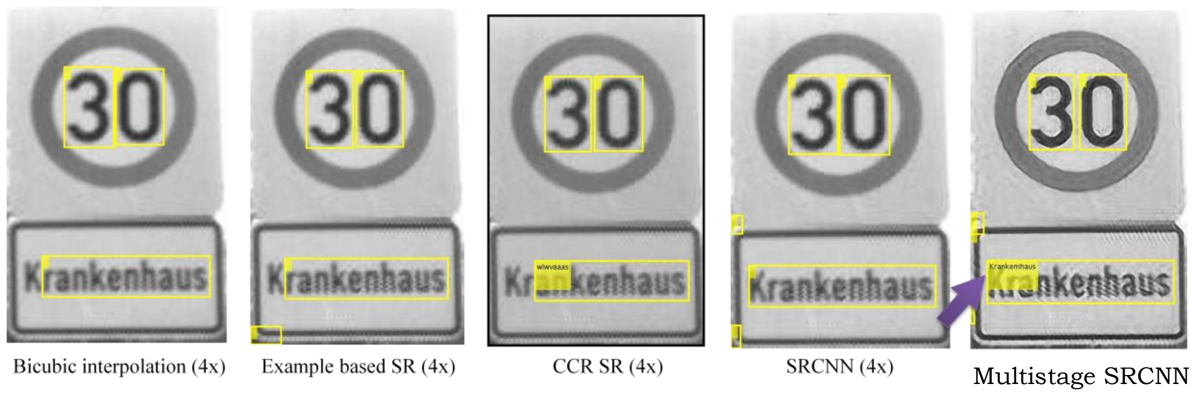


Figure 12: Comparison of proposed multi-stage SRCNN with other methods. Detection of text “Krankenhaus” is seen in SR image generated using multi-stage SRCNN.

The Figure 13 we can see the text being detected in below image which is super resolved using Multistage SRCNN method. We also can see the text 50 (the sign board data) is displayed in the detected text.

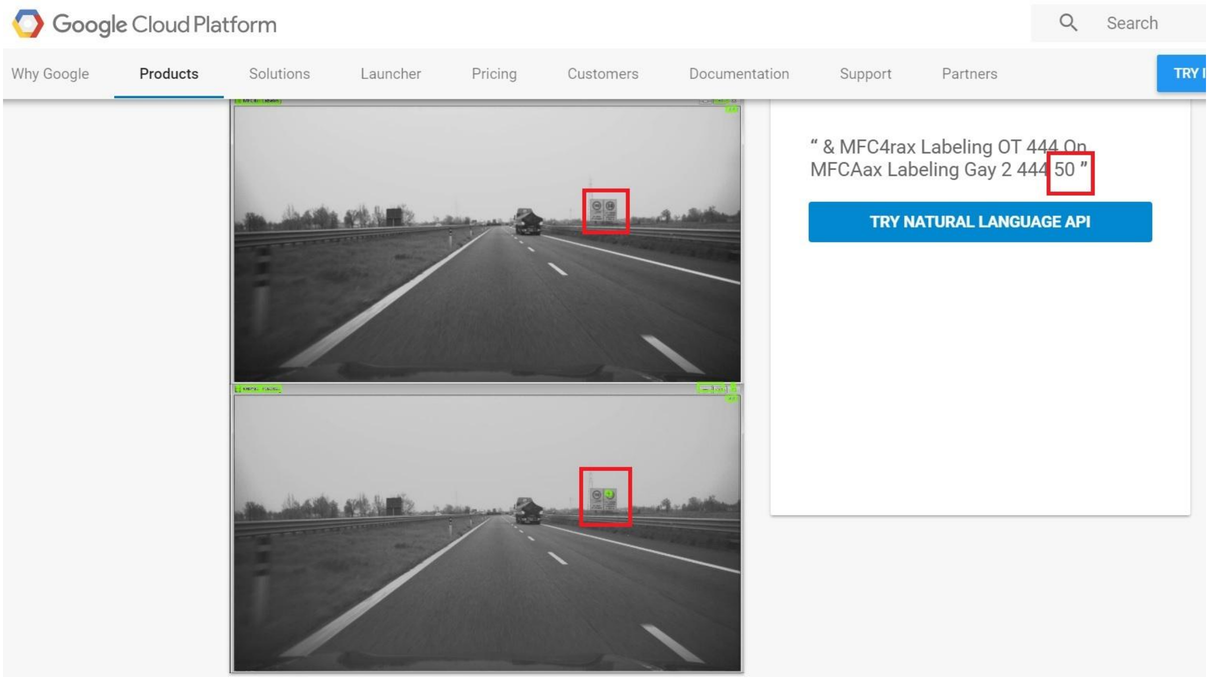


Figure 13: Text detection using Google Cloud Vision API. Below image is SR 4x using multistage SRCNN and above image is SR4x using bicubic interpolation.

3 Restoration of Motion Blur

In this section, we discuss different methods for restoration motion blur by estimating the length and angle discussed in literature. We also propose a method to estimate the length and angle of motion blur using nearest neighbour framework. We demonstrate the results of methods listed below:

- Steering Kernel for restoration of motion blur.
- Cepstral Method for Estimation of Length and Angle.

We demonstrate the results using different vision tools.

3.1 Methods tried for restoration of motion blur

3.1.1 Steering Kernel for restoration of motion blur.

As discussed in Section 2.1, the steering kernel assumes and handles limited blur.

- Steering Kernel assumes large motion compensation.
- Spatial and motion blur are embedded in the steering kernel.
- Initial estimate of the image gradients is made using gradient estimator (second order classical kernel regression).
- The estimate is used to measure the dominant orientation of the local gradients in the image.
- The orientation information is used to adaptively steer the local kernel to generate a deblurred/ super resolved image.

3.1.2 Cepstral Method for Estimation of Length and Angle [5]

The process of blurring is modelled as $g(x, y) = f(x, y) * h(x, y) + n(x, y)$,where $f(x,y)$ is the de-blurred image, $h(x,y)$ is the PSF to be estimated and $n(x,y)$ is the noise. A method for identifying linear motion blur is to compute the two-dimensional cepstrum of the blurred image $g(x,y)$. Cepstrum of an image is defined as the inverse Fourier transform of the logarithmic transform of the Fourier transform of the given image.

i.e → The cepstrum of $g(x, y)$ is given by

$$C\,(g(x, y)) = F^{-1}\,(\log\,|F(g(x, y))|).$$

An important property of the cepstrum is that it is additive under convolution. Thus, ignoring noise, we have

$$C\,(g(x, y)) = C\,(f(x, y)) + C\,(h(x, y)).$$

Biemond shows in [1] that $C\,(h(x, y)) = F^{-1}(\log\{|H(x, y)|\})$ has large negative spikes at a distance L from the origin. By the additivity of the cepstrum, this negative peak is preserved in C (g(x, y)), also at a distance L from the origin. If the noise level of the blurred image is not too high, there will be two pronounced peaks in the cepstrum, as shown in Figure 1. To estimate the angle of motion blur, draw a straight line from the origin to the first negative peak. The angle of motion blur is approximated by the inverse tangent of the slope of this line.

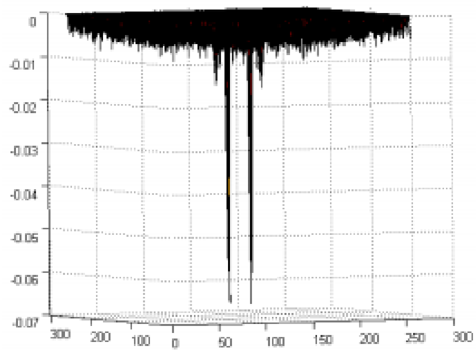


Figure 14: Cepstrum showing peeks

Algorithm:

1. Convert blurred RGB image (x,y,3) to gray level image f(x,y).
2. Perform Hann windowing over the image to remove boundary artefacts.
3. Compute the Fourier transform F(u, v) of step2 image.
4. Compute the log spectrum of F(u,v).
5. Compute the inverse Fourier transform of log spectrum.
6. Plot three dimensional plot of the cepstral function considering row index as x-axis, column index as y-axis, and the cepstral function value as the z-axis.

7. Find the largest two negative spikes from the plot and find slope of line along the x-y plane
8. Estimated angle is approximately inverse tangent of the slope of the line.
9. The square root of summation of squares of x intercept and y intercept gives the length through which blur is introduced

3.1.3 Proposed nearest neighbour framework for restoration of motion blur

We propose a nearest neighbour framework to estimate the length and angle that is responsible for the motion blur in the image. We use variants of first order features for estimation. We observe that the first order horizontal features vary as the length responsible for motion blur changes and the first order vertical features vary as the angle responsible for the motion blur changes. We propose a framework wherein we estimate length and angle separately using the horizontal and vertical first order features respectively.

In order to reduce the time taken for the estimation and also increase the accuracy, we use mean and variance of first order features as the features for estimation. Based on inferences from features, we estimate length and angle using nearest neighbour framework.

The algorithm for estimation of length responsible for motion blur is as explained below.

- Consider a set of images of different size.
- Apply synthetic blur on all the images with different lengths and angles to generate multiple blurred images.
- Extract first order horizontal features for all the blurred images.
- Find the variance and mean of each of the horizontal features extracted to create a mapping.
- For the test image, Extract first order horizontal features.
- Find the variance and mean of the horizontal features extracted for the test image.

- Use nearest neighbour algorithm to estimate the length from the mapping.
- The output would be the values of lengths that are possible according to the model in precedence.

Same algorithm is used to estimate the angle, with first order vertical features. Using the estimated length and angle, PSF is determined to perform blind de-convolution and restore the de-blurred image.

3.2 Results and Discussions

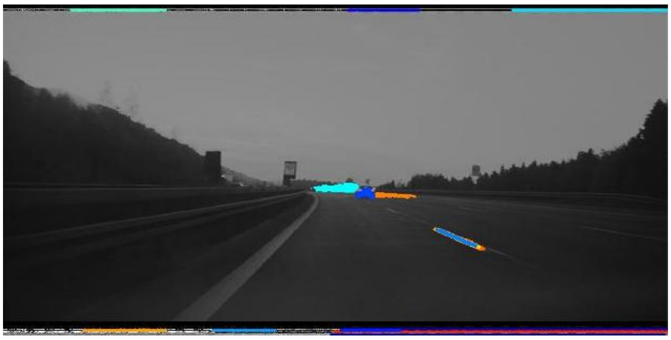
In this section, we demonstrate and compare the results of different methods of motion blur estimation with our proposed algorithm using different vision tools.

3.2.1 Results: Steering Kernel for restoration of motion blur.

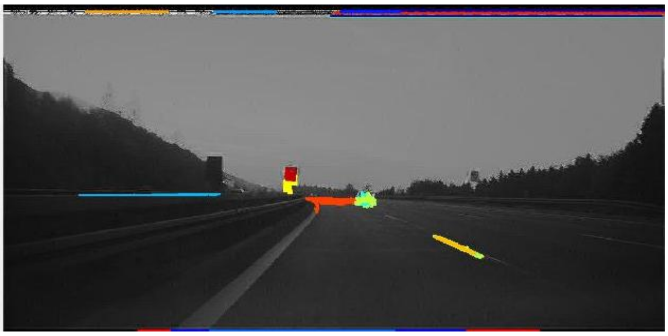
We experimented with steering kernel method with different size of kernels in space and time. The performance of steering kernel method for motion blur removal on the real image dataset was not satisfactory. We continued experimenting with other methods. The results are similar to shown in Figure 2.

3.2.2 Results: Cepstral Method for Estimation of Length and Angle.

In the Figure we see the signal board considered as MSER in the de-blurred image. We observe the performance of this method gives results in controlled conditions and for limited datasets. We experiment this method with different mask sizes and observe change in results. The prediction is not accurate in most of the cases.



(a)



(b)

Figure 15: Results of MSER-OCR on de blurred images (a) Blur Image (b) De-blurred image. We can see signal being considered as MSER.

3.2.3 Results: Nearest neighbour framework for restoration of motion blur

The Figures 16 to Figure 23 show the de-blurred images using proposed Nearest neighbour framework for restoration of motion blur for synthetic images generated for different combinations of length and angle.

(a) (b) (c)

Figure 16: Restoration of motion blur on synthetic data using proposed method,
(a) Ground truth image (b) Blurred image with Length = 7 and angle =8 (c)De-blurred image

(a) (b) (c)

Figure 17: Restoration of motion blur on synthetic data using proposed method, (a) Ground truth image (b) Blurred image with Length = 6 and angle =9 (c)De-blurred image

(a) (b) (c)

Figure 18: Restoration of motion blur on synthetic data using proposed method, (a) Ground truth image (b) Blurred image with Length = 4 and angle =2 (c)De-blurred image

(a) (b) (c)

Figure 19: Restoration of motion blur on synthetic data using proposed method, (a) Ground truth image (b) Blurred image with Length = 6 and angle =6 (c)De-blurred image

Figure 20: Restoration of motion blur on synthetic data using proposed method, (a) Ground truth image (b) Blurred image with Length = 8 and angle =10 (c)De-blurred image

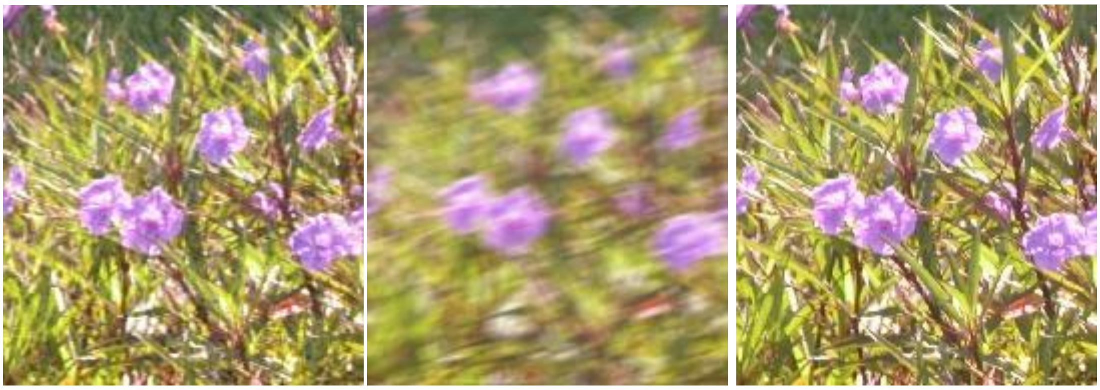


Figure 21: Restoration of motion blur on synthetic data using proposed method, (a) Ground truth image (b) Blurred image with Length = 12 and angle =18 (c)De-blurred image

Figure 22: Restoration of motion blur on synthetic data using proposed method, (a) Ground truth image (b) Blurred image with Length = 10 and angle =14 (c)De-blurred image

(a)

(b)

(c)

Figure 23: Restoration of motion blur on synthetic data using proposed method, (a) Ground truth image (b) Blurred image with Length = 6 and angle =12 (c) De-blurred image.

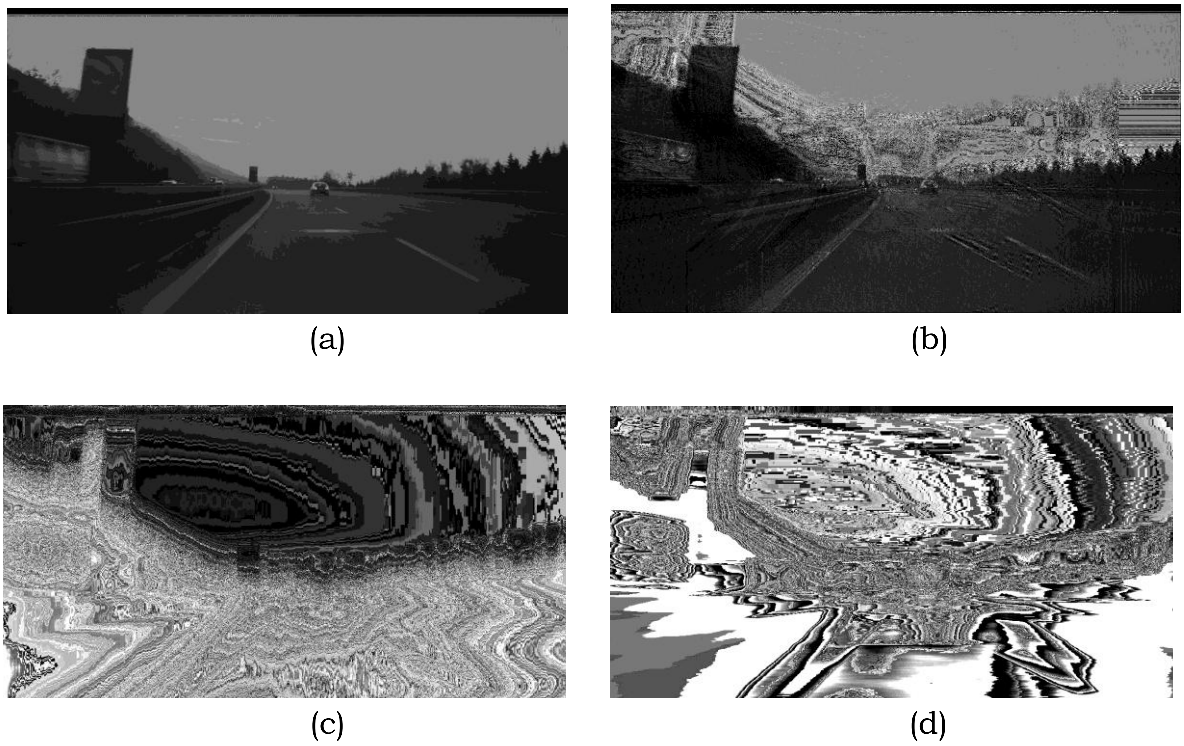


Figure 24 : Motion blur estimation on real image dataset (a) blurred image (b) de-blurred image (c) motion map of length (d) motion map of angle.

Figure 24 shows the result of nearest neighbour framework for restoration of motion blur on real and challenging scenario for motion blur estimation. The motion maps of the image in Figure 24 (a) is shown in Figure 24(c) and 24(d) for length and angle. We observe that the lent and angle predictions are intuitional using which the problem can be addressed. We try to restore the image using these estimates.

4. Contribution and Future Directions

4.1 Contributions

In this project,

- We propose a multi-stage SRCNN method to super resolve the image.
- We propose to estimate the motion blur in an image by modelling a nearest neighbour framework.
- We propose to demonstrate the quality of super resolved image using different vision tools.
- We demonstrate the results on real and synthetic dataset.

4.2 Future Directions

- The proposed algorithms of multi-stage SRCNN can be experimented with different training data sets, combinations of layers of convolution neural network and different learning at every stage.
- The blur removal framework can be modelled with learning different features to improve the results. Need to model a better restoration framework.

References:

1. Hiroyuki Takeda, Sina Farsiu, Peyman Milanfar, “Kernel Regression for Image Processing and Reconstruction”, IEEE Transactions on Image Processing, 2007.
2. K. I. Kim and Y. Kwon, “Example-based learning for single-image super-resolution”, in Proc. DAGM, 2008.
3. Yongbing Zhang, Yulun Zhang, Jian Zhang, “CCR: Clustering and Collaborative Representation for Fast Single Image Super-Resolution”, National Natural Science Foundation of China, 2014.
4. Chao Dong, Chen Change Loy, Kaiming He, Xiaoou Tang. “Learning a Deep Convolutional Network for Image Super-Resolution”, in Proceedings of European Conference on Computer Vision (ECCV), 2014.
5. F. Krahmer, Y. Lin, B. McAdoo, K. Ott, J. Wang, D. Widemann, “Blind Image Deconvolution: Motion Blur Estimation”, Information and Inference, a journal of the IMA, 2006.
6. Chen, Huizhong, et al. "Robust Text Detection in Natural Images with Edge-Enhanced Maximally Stable Extremal Regions." Image Processing (ICIP), 2011 18th IEEE International Conference on. IEEE, 2011.
7. Gonzalez, Alvaro, et al. "Text location in complex images." Pattern Recognition (ICPR), 2012 21st International Conference on. IEEE, 2012.
8. Li, Yao, and Huchuan Lu. "Scene text detection via stroke width." Pattern Recognition (ICPR), 2012 21st International Conference on. IEEE, 2012.
9. Neumann, Lukas, and Jiri Matas. "Real-time scene text localization and recognition." Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. IEEE, 2012.

10. Sudheer Babu and Sreenivasa Murthy. A Survey on the Methods of Super-Resolution Image Reconstruction. International Journal of Computer applications, 15:1_6, 2011.
11. Neil Patrick Del Gallego and Joel Ilao. Multiple-image super-resolution on mobile devices: an image warping approach. EURASIP Journal on Image and Video Processing, 2017(1):8, Jan 2017.
12. E. Faramarzi, D. Rajan, and M. P. Christensen. Unified blind method for multi-image super-resolution and single/multi-image blur deconvolution. IEEE Transactions on Image Processing, 22(6):2101_2114, June 2013.
13. N. Gayathri and P. L. Deepa. Multi-focus color image fusion using NSCT and PCNN. In 2016 International Conference on Communication Systems and Networks (ComNet), pages 173_178, July 2016.
14. Toshiyuki Kato, Hideitsu Hino, and Noboru Murata. Sparse coding approach for multi-frame image super resolution. CoRR, abs/1402.3926, 2014.
15. Soumya M and Anand M J. Interpolation based image super resolution by support-vector-regression. International Journal of Computer Science and Information Technologies (IJCSIT), pages 5010_5014, 2014.
16. Maria.P, Mohamed, Shengyong.C, and Mark.B. Super-resolution in practice: the complete pipeline from image acpyure to super resolved subimage using a novel frame selection method. Machine Vision and Applications, 23:441_459, 2012.
17. T Yoshida, T Murakami, and M Ikehara. Image super-resolution method based on non-local means and self similarity. in Intelligent Signal Processing and Communications Systems (ISPACS), International Symposium (2013), 2013.
18. Uma Mudenagudi, Subhashis Banerjee, and Prem Kumar Kalra. Spacetime super-resolution using graph-cut optimization. IEEE Trans. Pattern Anal. Mach. Intell., 33(5):995_1008, 2011.