

An Approach And Design To Enhance Video Denoising Algorithms

Dr Suresh M.B¹, Divyajyothi. G²

HOD, ISE, EWIT, Bangalore, Karnataka¹

M.TECH, ISE, EWIT, Bangalore, Karnataka²

Abstract— video denoising is highly desirable in many real world applications. It can enhance the perceived quality of video signals, and can also help improve the performance of subsequent processes such as compression, segmentation, and object recognition. Here the proposed algorithm is an effective strategy that aims to enhance the performance of existing video denoising algorithms. The idea is to denoise the noisy video as a 3-D volume using a given base 2-D denoising algorithm but applied from multiple views (front, top, and side views). A fusion algorithm is then designed to merge the resulting multiple denoised videos into one, so that the visual quality of the fused video is improved. And the extensive tests using a variety of base video-denoising algorithms show that the proposed method leads to surprisingly significant and consistent gain in terms of PSNR. Where the improvement over state-of-the-art denoising algorithms is often more than 2 dB in PSNR.

Keywords— polyview fusion, video denoising, video quality enhancement, PSNR.

I. INTRODUCTION

Video signals are often contaminated by noise during acquisition and transmission. Removing/reducing noise in video signals (or video denoising) is highly desirable, as it can enhance perceived image quality, increase compression effectiveness, facilitate transmission bandwidth reduction, and improve the accuracy of the possible subsequent processes such as feature extraction, object detection, motion tracking and pattern classification. Video denoising algorithms may be roughly classified based on two different criteria: whether they are implemented in the spatial domain or transform domain and whether motion information is directly incorporated. Spatial domain denoising is usually done with weighted averaging within local 2-D or 3-D windows, where the weights can be either fixed or adapted based on the local image content. To enhance perceived video quality, and to help improve the performance of subsequent processes, such as compression; segmentation; and object detection, recognition, and tracking [1]. Existing video denoising algorithms may be classified into 2-D and 3-D approaches. The simplest 2-D approaches denoise the video frame by frame by employing 2-D still-image denoising algorithms, for which well-known and state-of-the-art algorithms include spatially adaptive 2-D Wiener filtering (Wiener-2-D) [2], Bayes' least-square estimation based on the Gaussian scale mixture model (BLS-GSM) [3], nonlocal means [4], K-SVD [5], Stein's unbiased risk estimator-linear expansion of threshold (SURE-LET) [6], and block matching and 3-D transform shrinkage (BM3D) [7]. Since the correlation between neighboring frames is

completely ignored, these methods do not make use of all available information. Advanced 2-D approaches explore the correlation between adjacent frames. By incorporating motion compensation processes, state-of-the-art image denoising algorithms were extended to video, leading to the ST-GSM [8], and video SURE-LET [9] algorithms. In [10], multiple similar patches in neighboring frames that may not reside along a single trajectory are found. This is followed by transform- and shrinkage-based denoising procedures. In the video BM3D (VBM3D) method [11], similar patches in both intra and interframe are aggregated before a two-stage 3-D collaborative filtering algorithm is employed for noise removal. Three-dimensional Video-denoising schemes treat video sequences as 3-D volumes. These methods may operate in the space-time domain by adaptive weighted local averaging [12], 3-D order-statistic filtering [13], 3-D Kalman filtering [14], or 3-D Markov-model-based filtering [15]. They may also be applied in the 3-D transform domain, where soft/hard thresholding or Bayesian estimation is employed to eliminate noise, followed by an inverse 3-D transform that brings the signal back to the space-time domain [16]. Recently, 3-D-patch-based methods that achieved highly competitive denoising performance have also been investigated [17], [18]. To make best use of all available information, an ideal video-denoising algorithm would need to operate in 3-D. However, in the presence of significant motion, direct space-time 3-D filtering or 3-D transform-based approaches are difficult to effectively cover all motion-associated image content within local regions. On the other hand, 2-D denoising algorithms that use intra- and/or interframe information may be more efficient, but their performance is restricted by not taking full advantage of the neighboring pixels in all three dimensions simultaneously. Here, we propose a polyview fusion (PVF) scheme, where the same noisy video volume is denoised using 2-D approaches but from three different views, i.e., front, top, and side views. This is followed by a normalization procedure inspired by the structural similarity (SSIM) measure [19] and a fusion process based on local variance. By doing so, the advantage of 2-D approaches is utilized, whereas each pixel is denoised by its neighboring pixels from all three dimensions, thus providing a compromise between 2-D and 3-D approaches.

II. PROPOSED METHOD

A digital video signal can be expressed as a 3-D function $f(u, v, t)$ discrete in both space and time, where u and v are the horizontal and vertical spatial indices, respectively, and t is the time index. A video is typically played along the time axis. At any time instance $t=t_0$, the video is displayed as a 2-D front-view image $f(u, v, t_0)$, and the image changes for different values of t_0 . If we consider a video signal as 3-D

volume data, then it can also be viewed from the side or the top. This gives two other ways to play the same video, i.e., a sequence of 2-D top-view images $f(u_0, v, t)$ for different values of u_0 and a sequence of 2-D side-view images $f(u, v_0, t)$ for different values of v_0 .

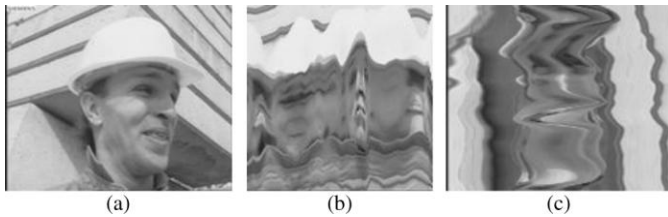


Fig. 1. Video signal observed from (a) front view, (b) side view, and (c) top view.

An example is shown in Fig. 1, where the rarely observed side- and top-view images demonstrate some interesting regularized spatiotemporal structures. Let x be an original noise-free video signal that is contaminated by additive independent zero-mean noise n with standard deviation σ_n resulting in a noisy signal

$$y = x + n$$

A video-denoising operator $D(\cdot)$ takes the noisy observation and maps it to an estimator of x , i.e.,

$$\hat{x} = D(y)$$

So that the difference between x and \hat{x} is as small as possible. The proposed PVF method relies on a base video-denoising algorithm. The base denoiser is applied to the same noisy signal y but from different views, resulting in multiple versions of denoised signals, i.e.,

$$z_i = D_i(y),$$

In our current work, $z = (z_1, z_2, z_3, \dots, z_N)$ because we have three different views, but in principle, the general approach also applies to the cases of less or more views, or multiple denoising algorithms. Fig. 2 shows sample denoised frames created by applying different denoising algorithms from three different views. It can be observed that the denoised frames have quite different appearances, even when the same denoising method is applied (from different views). Some image structures preserved in one of the views may be missing in the other views, and some artifacts that appear in one view may also be absent from another view. This suggests that the denoised frames from different views could complement each other, and fusing them (in appropriate ways) could potentially improve the denoising result. Let $z = (z_1, z_2, z_3, \dots, z_N)$ be a vector that contains all denoised results. Then, the final denoised signal \hat{x} is obtained by applying a fusion operator $F(\cdot)$ to z i.e.,

$$\hat{x} = d(y) = F(z) = F(D_1(y), D_2(y), \dots, D_N(y)).$$

In the case that the base denoisers D_i are predetermined, the remaining task is to define fusion rule. Before the fusion step, however, we first apply a normalization process to each z_i . This is inspired by the SSIM index [19], which has been shown to be a much better predictor of the perceived image quality than the mean squared error (MSE). Given two image patches, the SSIM index separates the similarity measure into the luminance, contrast, and structure components. Since the luminance and contrast (measured by mean intensity and standard deviation, respectively) of an image patch can be adjusted freely without changing its structure, we can improve the SSIM measure by adapting the luminance and contrast of each z_i to match those of x while maintaining its structure. Specifically, we compute

$$\hat{z}_i = \frac{\sigma_x}{\sigma_{z_i}}(z_i - \mu_{z_i}) + \mu_x$$

where μ_x and μ_{z_i} and σ_x and σ_{z_i} , denote the means and standard deviations of x and z_i , respectively. The computation in (1) requires the mean and standard deviation of x , which is not available. Fortunately, we can estimate them from noisy signal y using (1) and known noise properties (independence, zero mean and known standard deviation) by

$$\mu_x = \mu_y$$

And

$$\sigma_x = \sqrt{\sigma_y^2 - \sigma_n^2}$$

Where μ_y and σ_y^2 are the mean and variance of y , respectively. Our fusion rule is based on variance weighted averaging, which can be expressed as

$$\hat{x} = \frac{\sum_{i=1}^N \sigma_{z_i}^2 \hat{z}_i}{\sum_{i=1}^N \sigma_{z_i}^2}$$

This is determined by our empirical studies on the relationship between the variance and the quality of denoised video patches using state-of-the-art video-denoising algorithms. Specifically, for three given 3-D patches denoised by the same video denoising algorithm but from three different views, we compute their corresponding variances and PSNR values between the denoised and original patches. We then calculate the Spearman rank-order correlation coefficient (SRCC) between the three variance and three PSNR values. Table I shows the average SRCC values (over all patches) for nine video sequences denoised with four denoising algorithms. It can be seen that, although a fairly large variations are observed (depending on both denoising algorithm and video sequence), the correlations are all positive. This suggests that the patches of larger variances tend to have better image quality, thus justifying variance-based weighting.

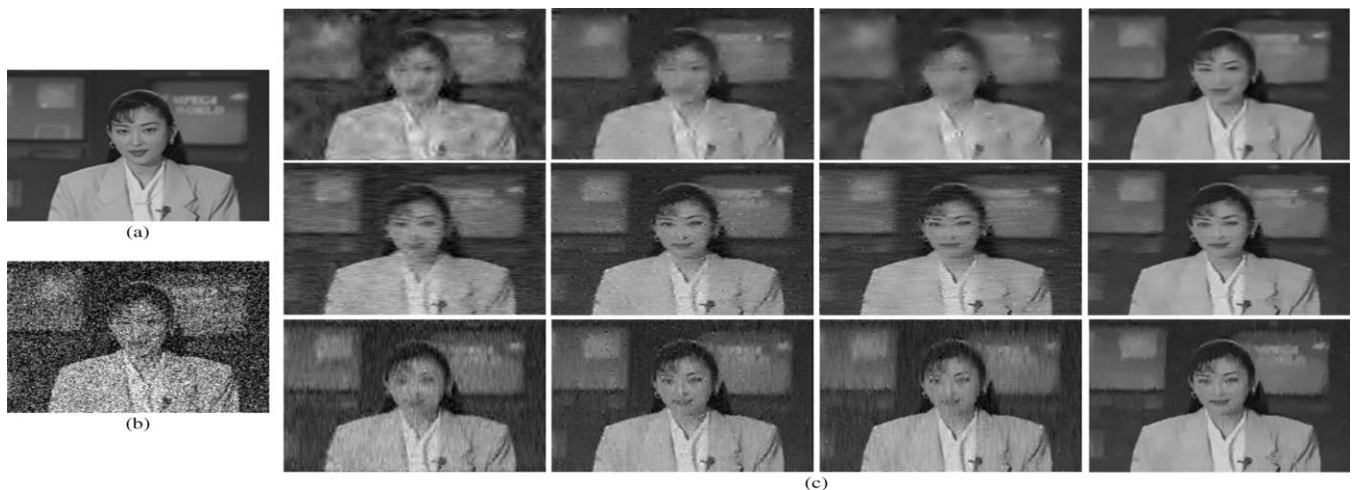


Fig. 2. Denoised frames from three different views using different denoising Igorithms. (a) Original frame. (b) Noisy frame with $n=50$ (c) (Left to right) Denoised frames by SURE-LET, BLS-GSM, K-SVD, and VBM3D. (Top to bottom) Denoised frames from front, top, and side views, respectively.

III .DESIGN CONSIDERATIONS

Design is one of the most important phases of software development. The design is a creative process in which a system organization is established that will satisfy the functional and non-functional system requirements. Large Systems are always decomposed into sub-systems that provide some related set of services. The output of the design process is a description of the Software architecture. Denoising technique can be classified according to BM3D, WEINER FILTER. The purpose of the design is to plan the solution of the problem specified by the requirements document. This phase is the first step in moving from problem to the solution domain. The design of the system is perhaps the most critical factor affecting the quality of the software and has a major impact on the later phases, particularly testing and maintenance. System design describes all the major data structure, file format, output as well as major modules in the system and their Specification is decided. In this, the system is broken into different modules, with a certain amount of dependency among them.

The system has the following modules:

- EXTRACT VIDEO IN MULTIPLE VIEWS (FRONT, TOP, SIDE)
- BASE FILTERS(BM3D)
- FUSION OF ALL THE VIEWS

In this project we are using bm3d has base filter Many video denoising methods have been proposed in the last few years. Prominent examples of the current developments in the .held are the wavelet based techniques. These methods typically utilize both the scarcity and the statistical properties of a

multiresolution representation as well as the inherent correlations between frames in temporal dimension. A recent denoising strategy, the non-local spatial estimation, has also been adapted to video denoising. In this approach, similarity between 2D patches is used to determine the weights in a weighted averaging between the central pixels of these patches. For image denoising, the similarity is measured for all patches in a 2D local neighborhood centered at the currently processed coordinate. For video denoising, a 3D such neighborhood is used. The electiveness of this method depends on the presence of many similar true-signal blocks. Based on the same assumption as the one used in the non-local estimation, i.e. that there exist mutually similar blocks in natural images, we proposed an image denoising method.

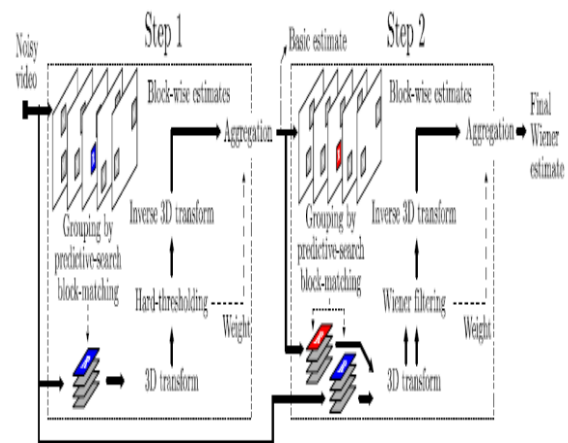


Fig 3 : BM3D block diagram

There, for each processed lock, we perform two special procedures grouping and collaborative filtering finds

Grouping mutually similar 2D blocks and then stacks them together in a 3D array that we call group. The benefit of grouping highly similar signal fragments together is the increased correlation of the true signal in the formed 3D array. Collaborative denoising takes advantage of this increased correlation to electively suppress the noise and produces estimates of each of the grouped blocks. That this approach is very electives for video denoising.

IV.SYSTEM ARCHITECTURE

The architecture for video denoising describes the details of the denoising the video. The operations performed are

- EXTRACT VIDEO IN MULTIPLE VIEWS (FRONT, TOP, SIDE)
- BASE FILTERS(BM3D)
- FUSION OF ALL THE VIEWS

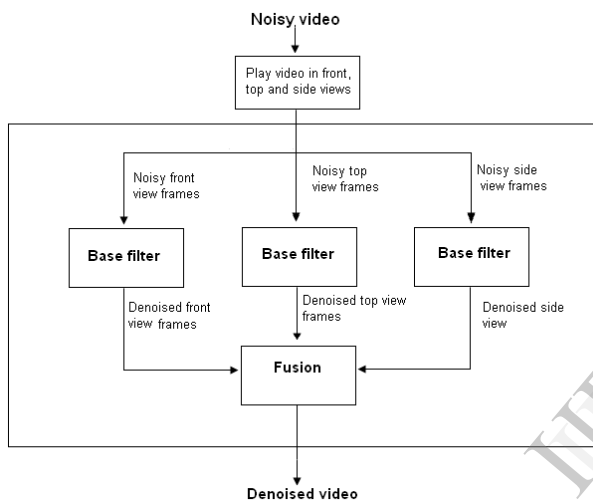


Fig 4 : system flow chart



Fig. 5. Comparison of one denoised frame from the "Akiyo" sequence with and without PVF. In the SSIM quality maps, brighter

pixels indicate higher SSIM values and, thus, better quality. (a1)–(e1) Wiener2-D, SURE-LET, BLS-GSM, K-SVD, and VBM3D denoised frames without PVF. (a2)–(e2) SSIM quality mapsfor (a1)–(e1). (a3)–(e3) Wiener2-D, SURE-LET, BLS-GSM, K-SVD, and VBM3D denoised frames with PVF. (a4)–(e4): SSIM quality maps for (a3)–(e3).

V.EXPERIMENTAL RESULTS

The proposed approach is tested on publicly available gray scale video sequences, which contain various content and rich motion styles. The sequences are of size 144 * 176 * 144 and are contaminated by independent zero-mean white Gaussian noise, where the standard deviation of the noise covers a wide range between 10 and 100. After the noisy sequences are denoised using a base denoiser along three different views, the noisy and denoised sequences are divided into 16 * 16 * 16 no overlap 3-D patches, within which sample means and variances are computed and employed in the normalization and fusion processes. The choices of nonoverlapping patches and size 16 are based on compromises between the denoising performance and complexity. In our simulations, there is no clipping of our- of-range values in the noise contamination and denoising processes.

All test sequences are in YCbCr 4:2:0 format and only the denoising results of the luma channel are reported here. Two objective criteria PSNR and SSIM are employed to evaluate the quality of the denoised video. Assume that x and y are the noise-free and denoised images, respectively, and L is the dynamic range of intensity values. Then

$$PSNR(x, y) = 10 \log_{10} \left(\frac{L^2}{MSE(x, y)} \right).$$

Where $C1$ and $C2$ are small positive constants to avoid instability when the means and variances are close to zero. This computation is applied at each location in the image using a sliding window that moves pixel by pixel across the image, resulting in an SSIM quality map, as demonstrated in Fig. 3. The SSIM value between two images is then computed as the mean of the SSIM map. Both PSNR and SSIM were computed on a frame-by-frame basis along the temporal direction and then averaged over all frames to yield the PSNR and SSIM values of the whole sequence. We test the proposed PVF method with diverse types of based denoisers, including the Wiener-2-D (using Mat lab Wiener2 function), VBM3D algorithms. The denoising computations are conducted using the default parameter settings of the code available to the public respectively.

Due to space limit, here, we only report the results of six sequences at five noise levels using six base denoising methods with and without PVF. The average improvement over nine test sequences is given in Table III. It can be observed that the proposed PVF approach leads to consistent performance gain over all base denoising algorithms, for all test video sequences, and at all noise levels. The gain is particularly significant at high noise levels, where the PSNR improvement could be 2 dB or higher upon the best video-denoising algorithms reported in the literature.

We also observe that the gain is reduced for video sequences with significant amount of large motion. Fig. 3 provides visual comparisons of the denoising results of one frame extracted from “Akiyo” sequence, for which the original and noisy frames are given in Fig. 2(a) and (b), respectively. Visual quality improvement by the proposed PVF approach can be easily discerned at various locations in the denoised frames. The observation is also verified by the SSIM quality map, which provides a useful indicator of local image quality variations. Furthermore, another experiment has been conducted to measure the computational complexity of the PVF operation and how it compares

The percentage of time spent on PVF ranges from 0.004% to 4.276% of the overall denoising process (where a base denoiser needs to be run three times and, thus, the overall process increases the computational cost by a factor of 3 or more). In conclusion, the complexity of the overall denoising algorithm mainly depends on the complexity of the base denoiser, and the PVF portion is mostly negligible.

TABLE I: AVERAGE PSNR AND SSIM IMPROVEMENT OVER ALL TEST SEQUENCES

Noise std (σ_n)	10	15	20	50	100
PSNR Improvement (dB)					
Wiener-2D	1.428	1.473	1.905	3.218	4.973
SURE-LET	1.882	2.050	2.143	2.472	2.780
BLS-GSM	1.848	1.980	2.068	2.248	2.007
K-SVD	1.748	1.817	1.853	1.895	1.772
ST-GSM	0.582	0.643	0.678	1.015	1.268
VBM3D	0.245	0.259	0.311	1.038	1.958
SSIM Improvement					
Wiener-2D	0.034	0.064	0.093	0.193	0.226
SURE-LET	0.024	0.036	0.047	0.094	0.141
BLS-GSM	0.048	0.023	0.030	0.065	0.081
K-SVD	0.013	0.020	0.026	0.048	0.049
ST-GSM	0.002	0.003	0.010	0.012	0.021
VBM3D	0.001	0.002	0.003	0.023	0.061

with the complexity of the base denoisers. The results are reported in Table IV, where the speed is measured in seconds based on Matlab implementations of the algorithms on a computer with Intel Core2 Duo CPU E8600 processor at 3.33 GHz. Although the implementations are not speed optimal, they give us a general idea about the amount of added complexities due to the PVF process. As can be observed, generally, the PVF procedure is of low complexity relative to the base denoising algorithms.

TABLE II
COMPUTATIONAL COMPLEXITY ANALYSIS

Base denoiser	One view denoising time (second)	PVF time (second)	PVF (%)
Wiener-2D	1.353	0.1813	4.276
SURE-LET	27.13		0.222
BLS-GSM	140.3		0.043
K-SVD	684.5		0.009
ST-GSM	1748		0.004
VBM3D	8.791		0.683

VI. CONCLUSION

A PVF approach is proposed to enhance video-denoising algorithms by fusing denoising results from multiple views. Our experiments demonstrate significant and consistent improvement over existing video-denoising methods. In practice, to apply PVF, one would need to store all video frames involved in the denoising and fusion processes in the memory. This may be a problem in practical systems, particularly when the video sequence is long. It is therefore preferable to divide long sequences into segments along the temporal direction and then denoise each segment independently. By adjusting the length of the segments, the memory requirement can be controlled. In the future, better denoising results may be obtained by incorporating more advanced denoising algorithms or by improving the fusion method. Although our current implementation only fuses the denoising results by the same base denoiser applied along three views, the general PVF approach facilitates fusing the results of any finite number of denoising algorithms. Two issues are critical to the success of this approach. First, the denoising algorithms need to be complementary to each other. Second, the fusion algorithm needs to select the best denoising result among many or optimally assign weights to multiple denoising results. In our current experiment, we observe that 2-D approaches from different views tend to be more complementary to each other than 3-D approaches, which have already considered the dependencies between neighboring pixels from all directions. Since the structural regularities exhibited in the top and side views are substantially different from those in the front view (as can be observed in Fig. 2), it is preferable to use different denoising methods that are best suited to the corresponding views before fusing the results. Currently, no denoising algorithm specifically tuned to denoise from top and side views has been developed. This gives us another interesting topic for future study.

ACKNOWLEDGMENT

I would like to take this opportunity to thank a lot of eminent personalities, without whose constant encouragement, this endeavor of mine would not have become a reality. Firstly, I would like to thank the VTU, BELGAUM, for having this paper as part of its curriculum, which gave me a wonderful opportunity to work on my research and presentation abilities and EWIT for providing me with such excellent facilities, without which, this seminar could not have acquired the shape it has now done. At the outset I would like to make a special mention of **Dr. Suresh.M.B**, Head of the Department of Information Science and Engineering, EWIT, BANGLORE who has guided me towards becoming technically more competent, but also for having taking the pains to provide me the necessary facility. I express our sincere thanks to **Dr.K.Channakeshavalu**, Principal of EWIT, for providing us the facilities. Finally, I would like to thank

all my friends and families for their constant support, guidance and encouragement.

REFERENCES

- [1] A. C. Bovik, Handbook of Image and Video Processing (Communications, Networking and Multimedia). Orlando, FL: Academic, 2005.
- [2] [Online]. Available: <http://www.mathworks.com/help/toolbox/images/ref/wiener2.html>
- [3] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain," *IEEE Trans. Image Process.*, vol. 12, no. 11, pp. 1338–1351, Nov. 2003.
- [4] A. Buades, B. Coll, and J. M. Morel, "Nonlocal image and movie denoising," *Int. J. Comput. Vis.*, vol. 76, no. 2, pp. 123–139, Feb. 2008.
- [5] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.
- [6] T. Blu and F. Luisier, "The SURE-LET approach to image denoising," *IEEE Trans. Image Process.*, vol. 16, no. 11, pp. 2778–2786, Nov. 2007.
- [7] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.
- [8] G. Varghese and Z. Wang, "Video denoising based on a spatiotemporal Gaussian scale mixture model," *IEEE Trans. Circuits Syst. Video Techn.*, vol. 20, no. 7, pp. 1032–1040, Jul. 2010.
- [9] F. Luisier, T. Blu, and M. Unser, "SURE-LET for orthonormal wavelet domain video denoising," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 6, pp. 913–919, Jun. 2010.
- [10] A. Buades, B. Coll, J. M. Morel, and D. Matématiques, "Denoising image sequences does not require motion estimation," in *Proc. IEEE Conf. AVSS*, 2005, pp. 70–74.
- [11] K. Dabov, A. Foi, and K. Egiazarian, "Video denoising by sparse 3-D transform-domain collaborative filtering," in *Proc. 15th Eur. Signal Process. Conf.*, Poznan, Poland, Sep. 2007, pp. 145–149.
- [12] M. Ozkan, M. Sezan, and A. Tekalp, "Adaptive motion-compensated filtering of noisy image sequences," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 3, no. 4, pp. 277–290, Aug. 1993.
- [13] G. Arce, "Multistage order statistic filters for image sequence processing," *IEEE Trans. Signal Process.*, vol. 39, no. 5, pp. 1146–1163, May 1991.



Dr. Suresh M.B received M.Tech degree from Visvesvaraya Technological University in Computer Science & Engineering and Ph.D from the University of Allahabad, in the area image processing, from the Department of Computer Science. He is having more than 15 years of experience in academics and has published several

papers at various National/International Journals/Conferences. He is a member of ISTE, IACSIT. Currently he is heading the Department of Information Science & Engineering, East West Institute of technology, Bangalore. His research interests include networking, image processing and wireless communication.



Ms. Divyajyothi. G received B.E. in Information Science and Engineering under the viswesvaraya Technological University, Belgaum at East West institute of Technology and Perusing M Tech in Computer Networking & Engineering under the Viswesvaraya Technological University, Belgaum, Karnataka in the Department of information science & Engineering at East West Institute of Technology, Bangalore.