Color Video Denoising Based on Combined Interframe and Intercolor Prediction

Xiaowen Ke. Author, MEng student of Concordia University

***Abstract*—Here I implemented a novel method in denoising video. Here I use the method call CIFIC. The process will develop based on inter frame and inter color correlation of the frames. Initially I read the video. The video will be converted as frames. It is one of the many still images which compose the complete moving picture. RGB color space exploits both the interframe and intercolor correlation in color video signal directly by forming multiple predictors for each color component using intercolor correlation to all three color components in the current frame as well as the motion-compensated neighboring reference frames. In that frames we separate the color channel from each frames, the separated color frames passed to the next process. Here we add the Gaussian noise with the variance of 0.005 to each frame. Added noise frame is then passed to the denoising process. Then we use the CIFIC algorithm for denoising. The algorithm will compute the interframe and intercolor correlation. Intercolor correlation will calculated in the current frame. And then LMMSE filtering will apply to filter the noise in each frame to corresponding color channel. After filtering the noise of the separated color channel will reconstruct to produce the frame. And finally the filtered frame will reconstruct as a video.**

***Index Terms*—intercolor, interframe and denoising.**

I. Introduction

Video signals are more and more popular nowadays with the increasing of multimedia technology and the internet applications. However, because of the imperfect process of video acquisition, storage and transmission technique, videos are always noisy. These noises will affect people's watching feelings. Furthermore, too much noise will affect the performance of the video compression, analysis, object tracking and pattern recognition [1] of the later video processes.

Therefore, a robust video denoising technique is very necessary in the video processing system. From lots of previous research, we often model the overall noise as an additive Gaussian white noise, which is independent of the original signals [1].

In video and image processing area, there are abundant varying efforts have been proposed. There are mainly two types of denoising approaches. One is based on the pixels while the other is operated in the frequency domain.

In pixel-based algorithms, the pixel intensity is the major object of the operation. Such as the famous and popular nonlocal technique, which is developed according to the theory of redundancy of the image. Instead of averaging the neighboring pixel directly, NLM algorithm aims at eliminating the neighboring pixels dissimilar to the pixel under study by assigning a weight to each neighboring pixel. Another famous algorithm called Multi Hypothesis Motion-Compensated Filter (MHMCF) is combined with the motion estimation (ME) and the Linear Minimum Mean Square Error (LMMSE) filter.

Unlike the pixel-based algorithms, frequency-domain algorithms firstly transfer the image into the frequency domain, then do the process in the transformation domain. The advantage of this kind of approach is that it can better utilize the characteristics of noise. Among these methods, wavelet transform is a very common linear transform methods. Besides, a famous algorithm called VBM3D denoising algorithm and further VBM4D are proposed, which needs 3-D and 4-D data respectively.

However, most of the existing denoising algorithms mentioned above is developed based on the grayscale video. A very intuitive idea of dealing color video is to directly apply the denoising to each color channel. Nevertheless, some researches indicates that there is strong relationship between these color components. Therefore, this characteristic is non-ignorable during the denoising of color video.

In this project, I aim at implementing a novel denoising scheme called CIFIC (standing for combined interframe and intercolor prediction). It utilizes combined interframe and intercolor prediction in the LMMSE filtering to enhance the denoising performance. Two additional features of CIFIC include a joint-RGB ME to generate a single motion field for the three color components simultaneously, as well as the detection and remedy of the ill condition in the LMMSE weight determination.

II. Related works

In order to better understand this proposed video denosing algorithm and have a more comprehensive understanding of video denoising area, I briefly read some related papers, which help me to better understand some details of video denoising from different aspects.

The first related paper is the paper proposing the CIFIC video denoising algorithm, which is the main reference of my project. In paper [1], an advanced color video denoising scheme which we call CIFIC based on combined interframe and intercolor prediction is proposed. CIFIC performs the denoising filtering

in the RGB color space, and exploits both the interframe and intercolor correlation in color video signal directly by forming multiple predictors for each color component using all three color components in the current frame as well as the motion-compensated neighboring reference frames.

The second reference paper [2] gives me some knowledges about motion compensation which is a very important theory in video denoising area. As we know, denoising module is required by any practical video processing systems. Most existing denoising schemes are spatial-temporal filters which operate on data over three dimensions. However, to limit the number of inputs, these filters only utilize one reference frame and cannot fully exploit temporal correlation. In this paper, a recursive temporal denoising filter named multi-hypothesis motion compensated filter (MHMCF) is proposed. To fully exploit temporal correlation, MHMCF performs motion estimation in a number of reference frames to construct multiple hypotheses (temporal predictions) of the current pixel. These hypotheses are combined by weighted averaging to suppress noise and estimate the actual current pixel value. Based on the multi-hypothesis motion compensated residue model presented in this paper, we investigate the efficiency of MHMCF, and some numerical evaluations are revealed. Experimental results show that MHMCF demonstrates quite well denoising performance while the inputs are much fewer than spatial-temporal filters. Moreover, as a purely temporal filter, it can well preserve spatial details and achieve satisfactory visual quality.

The third reference paper [3] is also about the motion compensation. A novel framework of the motion-compensated 3-D wavelet transform (MC3DWT) for video denoising is presented in this paper. The motion-compensated temporal wavelet transform is first performed on a sliding window of video frames consisting of previously denoised frames and the current noisy frame. The 2-D spatial wavelet transform is then performed on the temporal sub-band frames, thus realizing a 3-D wavelet transform. Any of established wavelet-based still image denoising algorithms can then be applied to the high-pass 3-D sub-bands. The operation of the inverse 2-D spatial wavelet transform followed by the inverse temporal wavelet transform reconstructs the video frames in the buffer. The denoised current frame may be used as an output for real-time processing; meanwhile, the past frames can be updated, one of which may be used as a delayed output for post-processing or for real-time processing that allows some amount of delay. The proposed MC3DWT framework integrates both the spatial filtering and recursive temporal filtering into the 3-D wavelet domain and effectively exploits both the spatial and temporal redundancies. Experimental results have demonstrated a superior visual and quantitative performance of the proposed scheme for various levels of noise and motion.

III. THEORY OF THE IMPLEMENTATION OF CIFIC

In my project, I mainly implement the proposed CIFIC algorithm from the reference paper [1]. There are several steps to achieve the system. The block diagram (figure 1) describes the entire system step by step. First, I need to do the pre-processing which converts the video into separated frames. Second, noise is generated and added to the signal. Third, each color frame is separated into Red, Green and Blue three grayscale channels. Then, the CIFIC is applied to the video. Finally, I calculate the cPSNR in order to analyzing the performance of the system objectively.

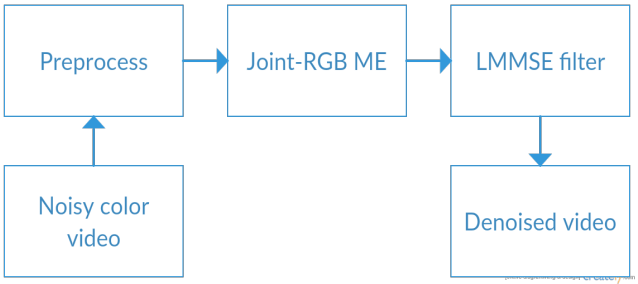

Figure 1 Block diagram of the entire system.

A. Pre-processing

In the pre-processing part, firstly I read the video. And the video was converted frame by frame. We know that video frame is one of the many still (or nearly so) images which compose the complete moving picture. The frame is composed of picture elements just like a chess board. Each horizontal set of picture elements is known as a line. The picture elements in a line are transmitted as sine signals where a pair of dots, one dark and one light can be represented by a single sine. The product of the number of lines and the number of maximum sine signals per line is known as the total resolution of the frame. Higher the resolution the more faithful the displayed image to the original image.

B. Noise generation

Then, the next step is to add the noise to each frame. As we know, image noise is random (not present in the object imaged) variation of brightness or color information in images, and is usually an aspect of electronic noise. It can be produced by the sensor and circuitry of a scanner or digital camera. The original meaning of "noise" is a remaining unwanted signal occurs in image. The standard model of amplifier noise is additive, Gaussian, independent at each pixel and independent of the signal intensity. Gaussian noise is statistical noise that has its probability density function equal to that of the normal distribution, which is also known as the Gaussian distribution. In other words, the values of the noise are Gaussian-distributed. A special case is white Gaussian noise, in which the values at any pair of times are identically distributed and statistically independent (and hence uncorrelated). In applications, Gaussian noise is most commonly used as additive white noise

to yield additive white Gaussian noise. Therefore, in my simulation, I added the Gaussian noise with variance of 0.005.

C. RGB separation

In this step, I separate the color video into red, green and blue grayscale channels since we apply the denoising algorithm to each grayscale channel but not color video directly. Generally, the color image contains three color channels. That is Red, Green and Blue. RGB channels roughly follow the color receptors in the human eye, and are used in computer displays and image scanners. If the RGB image is 24-bit, each channel has 8 bits, for red, green, and blue in other words, the image is composed of three images, where each image can store discrete pixels with conventional brightness intensities between 0 and 255. If the RGB image is 48-bit, each channel is made of 16-bit images. Each image is subjected to color component separation. Here we separate each image to have three components such as R, G and B. This is an additive color system based on tri-chromatic theory which is often found in systems using a CRT to display images. RGB is easy to implement but non–linear with visual perception. It is device dependent and specification of colors is semi–intuitive. RGB is very commonly being used in virtually in every computer system as well as television system, video system etc.

D. Applying CIFIC denoising method

Here we use CIFIC method which use the LMMSE filtering algorithm, specially designed for color video signal, utilizes combined interframe and inter-color prediction in the LMMSE filtering to enhance the denoising performance. Note that the LMMSE formulation in CIFIC can be written in the vector form like the second strategy. Two additional features of CIFIC include a joint-RGB ME to generate a single motion field for the three color components simultaneously. Initially we take the noise free red, green, blue component from the denoised frame. Then it considers the noisy frame color component of red, green, blue. The interframe color predictor, which will use to estimate the noise in the current pixel. Then the value of the current pixel will be changed by the mean value of the neighboring pixel. Likewise, the noise will be removed in the all the frames. Finally, the denoised channels of each frame are generated.

CIFIC mainly consists of two steps to denoise every block: joint-RGB motion estimation and motion-compensated LMMSE filtering. The current frame is divided into blocks with size of $K_x \times K_y$. In the motion estimation part, we calculate the trajectory for each block including three RGB components.

ME and MC (first part): aims at finding the temporal correspondence. We need to find the matched block from the same frame and the reference frames. The method to find the matched blocks is to apply motion estimation. However, there are some drawbacks when applying ME to separated color components: one is the huge computation burden and the other is that the SAD (Sum of Absolute Difference) and SSD (Sum of Squared Difference) are inapplicable in this case. Therefore, a joint-RGB ME is proposed. An block mismatch measurement is defined as

$$SSD_c(mv) = \sum_{x \in B}\big[\alpha_R\big(R_c ur(x) - R_r ef(x+mv)\big) + \alpha_G\big(G_c ur(x) - G_r ef(x+mv)\big) + \alpha_B(B_c ur(x) - B_r ef(x+mv))\big]^2 \quad (1)$$

where x denotes pixels in the current block, *cur* and *ref* indicate the current and reference frames respectively. Here α_R, α_G and α_B act as a weight for different color components which are very important in the next step.

$$\alpha_R = \frac{\sigma_{nR}^{-2}}{\sigma_{nR}^{-2}+\sigma_{nG}^{-2}+\sigma_{nB}^{-2}} \quad (2)$$

$$\alpha_G = \frac{\sigma_{nG}^{-2}}{\sigma_{nR}^{-2}+\sigma_{nG}^{-2}+\sigma_{nB}^{-2}} \quad (3)$$

$$\alpha_B = \frac{\sigma_{nB}^{-2}}{\sigma_{nR}^{-2}+\sigma_{nG}^{-2}+\sigma_{nB}^{-2}} \quad (4)$$

Then I move to second step, LMMSE filtering. In this step, I formulate the color noise removal by the LMMSE filter. Assuming M color reference frames are available, the denoised estimate of every color component of one pixel is constructed as a linear combination of *(3M + 3)* observations [1].

$$\hat{y}_R = W_R^T(z - \bar{z}) + \bar{y}_R \quad (5)$$

$$\hat{y}_G = W_G^T(z - \bar{z}) + \bar{y}_G \quad (6)$$

$$\hat{y}_B = W_B^T(z - \bar{z}) + \bar{y}_B \quad (7)$$

Here $\hat{y}_R$, $\hat{y}_G$ and $\hat{y}_B$ are denoised pixels in red, green and blue channel respectively.
Where W_R, W_G and W_B are the weighting vectors and:

$$z = [z_R, z_G, z_B, p_{1R}, p_{1G}, p_{1B}, \dots, p_{MR}, p_{MG}, p_{MB}] \quad (8)$$

$$\bar{z} = [\bar{z}_R, \bar{z}_G, \bar{z}_B, \bar{p}_{1R}, \bar{p}_{1G}, \bar{p}_{1B}, \dots, \bar{p}_{MR}, \bar{p}_{MG}, \bar{p}_{MB}] \quad (9)$$

where z_R, z_G and z_B are the noisy red, green and blue components of a color pixel in current block [1] and p_{MR}, p_{MG} *and* p_{MB} are the pixels in different color components of reference frames.

As $(z - \bar{z})$ contains interframe predictors as well as intercolor predictors, CIFIC exploits both the interframe and intercolor correlation directly.

E. PSNR and MSE

The generated denoised frame will reproduce as the color frame. It means the separated color channels of the frame are combined. It is the process of rearranging the frames. After producing the denoised video, we analyze the image by the utilize of PSNR and MSE. PSNR stands for peak signal noise ratio, and MSE stands for Mean Squared Error.

PSNR is ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. Because many signals have a very wide dynamic range, PSNR is usually expressed in terms of the logarithmic decibel scale.

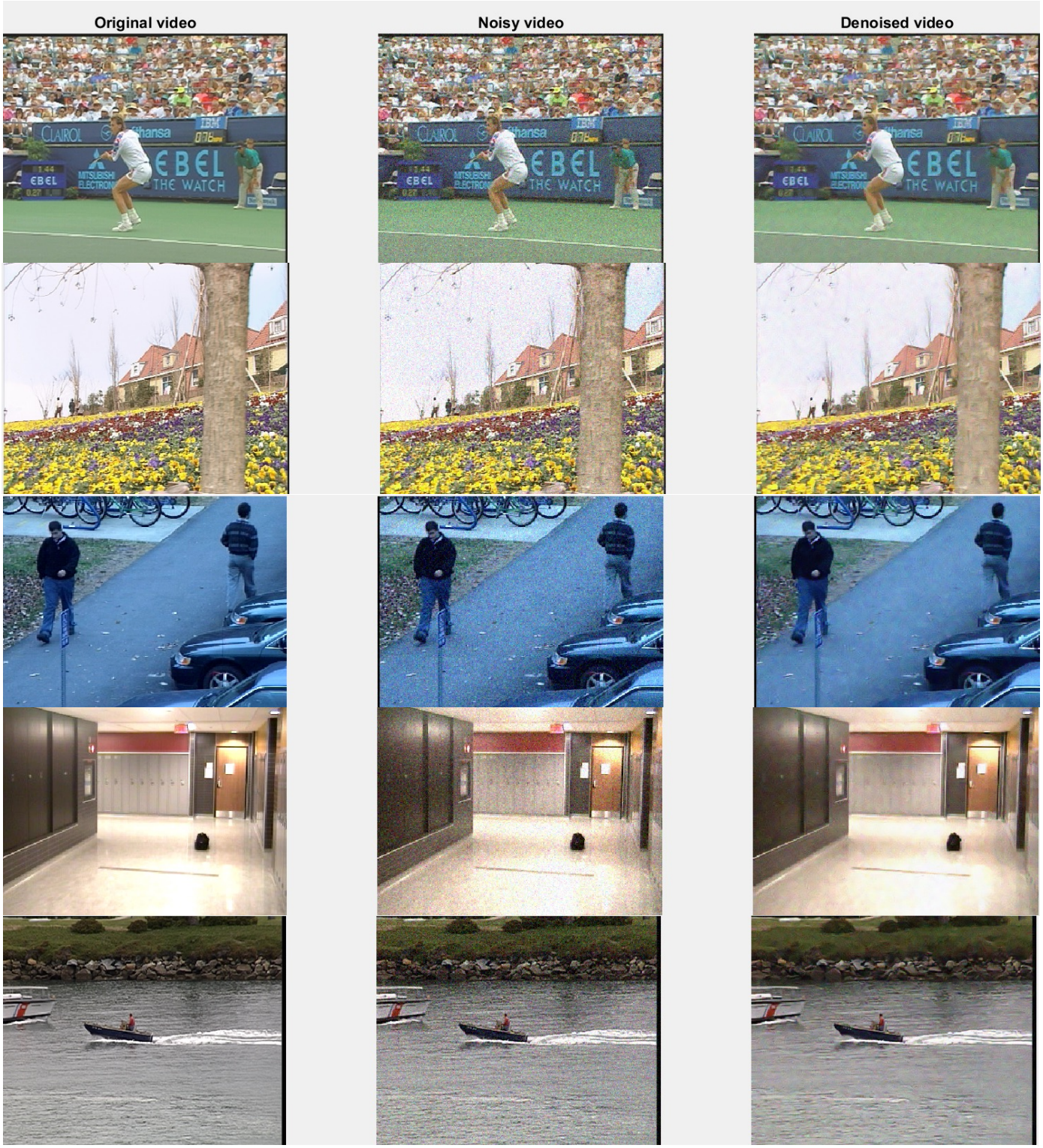

Figure 2 Simulation results (comparison between original noisy and denoised video) in the order (from top to bottom) of "stefan", "flower", "survey", "CUJul" and "coastguard".

The mean squared error (MSE) of an estimator is one of many ways to quantify the difference between values implied by an estimator and the true values of the quantity being estimated. MSE is a risk function, corresponding to the expected value of the squared error loss or quadratic loss. MSE measures the average of the squares of the "errors." The error is the amount by which the value implied by the estimator differs from the quantity to be estimated.

Figure 3 cPSNR comparison in the order (from top to bottom) of "stefan", "flower", "survey", "CUJul" and "coastguard".

IV. Results and Discussion

In my project, I used Matlab to implement the algorithm and do the simulation. About the Matlab coding, I set up a function *Intercorr,* which is the core of this algorithm. In the simulation part, I choose 5 different videos from the course website. For the performance of the proposed algorithm, I do the evaluation by subjective and objective ways. For subjective evaluation, PSNR is a common measurement. Here I used the cPSNR which is defined as

$$cPSNR = 10log(\frac{255^2}{(3K)^{-1}\sum_{c=R,G,B}\sum_x(y(x)-\hat{y}(x))^2}) \quad (10)$$

In the simulation part, the first step of implementing this algorithm is to calculate the joint-RGB motion estimation to find the temporal correspondence between the current frame and the reference frames. Then determine the weight for different color components as I mentioned in (2), (3) and (4). Then I used these weights to do the calculation in the next step LMMSE filtering.

For the analysis part, I mainly did two things. One is get the denoising results for each video. The second is to calculate the cPSNR for each frame of each video. Besides, in order to better analyze the denoising effect of this algorithm for different area (high frequency and low frequency), I also selected some homogeneous areas and the area with lots of details to do the comparison by calculating their cPSNR respectively.

As we can see from the results from figure 2, it is the visual comparison between the original video, noisy video and denoised video, which I take one frame of each video to demonstrate the denoising effect. Overall objectively speaking, the *cPSNR* of each video is over 30 except the video *"flower"*. And the range of *cPSNR* of each video is less than 2, which represents a good stability of the proposed CIFIC system.

Subjectively speaking, the result images show that the proposed CIFIC method can remove the noise effectively. For the homogeneous area, takes the ground area of the first video *"stefan"* for example, it is smoother than the noisy image and there is only few noise left. In this area, the CIFIC acts more like a normal averaging filter. However, it is a basic concept that the averaging filtering always has a severe blurring effect in the high frequency area resulting in losing details.

But for this CIFIC method, as can be seen from the result, it also shows a good performance in protecting the small details in this area. For example, in the area with some structures, take the advertise of the first video *"stefan"* for example, we still can recognize the letters. Furthermore, in the area with lots of some details, for example the auditorium, we can not see apparent blurring effect and subjective very close to the original image. The same result can be seen in the second video *"flower"*. On the flower area, we cannot see apparent difference from the original image. Besides, for the region with moving object especially fast moving object, it is a fact that it is the most challenging area for lots of denoising algorithm. There is a trade-off between temporal resolution and spatial resolution for many methods. However, in my project, the CIFIC demonstrate a good denoising quality in this area. Since it combines the interframe and intercolor correlation, it exploits more information of the video, therefore it is more accurate. As we can see from the moving object of these videos except *"flower"* we can not see much noise as well as the ghosting effect in these areas.

In order to have a deeper analyze and get a more precise objective measurement and comparison for the homogeneous and sharp areas, I set an extra part to the simulation. The purpose of this work is to verify the good denoising effect in the high frequency area subjectively. I divided the denoised image into several blocks with a size of 8×8. Then I calculate the variance of each blocks. For those who has a high variance, we can consider it as a textured region. In contrast, it is a homogeneous region if it has a low variance. After selecting the typical blocks, I calculated the cPSNR for these blocks respectively to compare the difference of denoising effect between homogeneous area and textured area.

Table 1 cPSNR comparison between flat and textured area.

	Flat area	*Textured area*
"stefan"	*32.5163*	*31.5178*
"flower"	*29.3237*	*28.6752*
"survey"	*38.1224*	*33.7112*
"CUJul"	*32.5712*	*30.1093*
"coastguard"	*33.7623*	*32.6540*

The result (*Table* 1) shows that the value of cPSNR of textured region is a little bit lower than it of the flat area. This is also acceptable. But it looks subjectively better than the flat area. There are mainly two reasons. One is that the resolution of the original video is low, therefore even for noise-free video we can not identify the details in the auditorium very well. Another more important reason is that human visual system acts as a low-pass filter. It is more sensitive to low frequency components, as a result we can recognize the noise in the flat area even if the noise level is low. In contrast, it is less sensitive to the high frequency components. The theory supporting this is show as below. The human eye can be modelled by

$$h(x,y,t) = \begin{cases} \frac{1}{T_xT_yT_e}, |x| < \frac{T_x}{2}, |y| < \frac{T_y}{2}, t \in (0,T_e) \\ 0 \qquad\qquad otherwise \end{cases} \quad (11)$$

where T_x, T_y are the horizontal and vertical size of the human eye aperture and the following results shows that the human visual system acts as a low pass filter.

$$H(f_x,f_y,f_t) = H_x(f_x)H_y(f_y)H_t(f_t) = \frac{\sin(\pi f_xT_x)}{\pi f_xT_x}\frac{\sin(\pi f_yT_y)}{\pi f_yT_y}e^{-j\pi f_tT_e}\frac{\sin(\pi f_tT_e)}{\pi f_tT_e} \quad (12)$$

V. Conclusion and Future Work

In this project, I simply implement the proposed denoising method CIFIC of reference paper [1]. This algorithm utilizes the interframe as well as intercolor correlation. That makes it more accurate but increases the computational complexity. For implementing this algorithm, there are mainly two steps: Joint-

RGB motion estimation and LMMSE filtering. Then, I used Matlab to do the simulation and added a small extra part to verify the denoising effect on particular area. In terms of experimental results, I analyze the denoising effect from both subjective and objective aspects. The result shows the effectiveness of this algorithm.

However, there is still some drawbacks. For example, the computation burden. Therefore, in the future, I should pay more attention to simplify the algorithm and the code. Furthermore, I should spend some time to do some comparisons with other common denoising algorithm and know how to apply these methods to different cases.

REFERENCES

[1] Jingjing Dai, Oscar C. Au, Chao Pang, and Feng Zou, "Color Video Denoising Based on Combined Interframe and Intercolor Prediction", *IEEE Trans. Circuits Syst. Video Technol.,* vol. 23, no. 1, pp. 128-141 Jan. 2013.

[2] L. Guo, O. C. Au, M. Ma, and Z. Liang, "Temporal video denoising based on multihypothesis motion compensation," *IEEE Trans. Circuits Syst. Video Technol*., vol. 17, no. 10, pp. 1423–1429, Oct. 2007.

[3] S. Yu, M. O. Ahmad, and M. N. S. Swamy, "Video denoising using motion compensated 3-D wavelet transform with integrated recursive temporal filtering," IEEE Trans. Circuits Syst. Video Technol., vol. 20, no. 6, pp. 780–791, Jun. 2010.