OPTICAL REVIEW Vol. 21, No. 5 (2014) 509–515

High-Speed Video Capture by a Single Flutter Shutter Camera Using Three-Dimensional Hyperbolic Wavelets

Kuihua HUANG*, Jun ZHANG, and Jinxin HOU

College of Information System and Management, National University of Defense Technology,
Changsha, Hunan 410073, P. R. China

(Received March 4, 2014; Accepted June 19, 2014)

Based on the consideration of easy achievement in modern sensors, this paper further exploits the possibility of the recovery of high-speed video (HSV) by a single flutter shutter camera. Taking into account different degrees of smoothness along the spatial and temporal dimensions of HSV, this paper proposes to use a three-dimensional hyperbolic wavelet basis based on Kronecker product to jointly model the spatial and temporal redundancy of HSV. Besides, we incorporate the total variation of temporal correlations in HSV as a prior knowledge to further enhance our reconstruction quality. We recover the underlying HSV frames from the observed low-speed coded video by solving a convex minimization problem. The experimental results on simulated and real-world videos both demonstrate the validity of the proposed method.

Keywords: high-speed video, flutter shutter camera, 3D hyperbolic wavelets, Kronecker product, total variation

1. Introduction

High-speed video (HSV) camera has many applications in scientific research, industrial detection, safety studies, military, etc. Due to its huge memory bandwidth requirement, HSV camera needs specialized readout circuit. In addition, since the exposure interval of each frame in HSV is very small, HSV camera requires high light sensitivity sensors to ensure each frame is above the noise bed. Both of the two factors result in a very expensive price of a HSV camera. Despite their high costs, HSV cameras are still limited in achieving simultaneous high spatial-temporal resolution, because current fast mass data storage devices do not have high enough write speed to continuously record HSV at high spatial resolution.[1)]

Recent advances in computational imaging and compressive sensing (CS) pave a new way for the development of HSV system and have led to a series of creative devices and models. One approach is to use multiple cameras. Wilburn et al. built a dense array of 100 low-speed cameras to recover a 1000 fps video.[2)] Shechtman et al. achieved spatial-temporal super-resolution by using multiple cameras with staggered exposures.[3)] Agrawal et al. proposed to use N low-speed flutter shutter cameras to recover a video performing an N times temporal resolution.[4)] Wu and Pournaghi also used multiple flutter shutter cameras to construct a coded video acquisition system.[1)] While HSV system using multiple cameras can produce very high-quality result, it suffers from many hardware challenges, including the camera calibration problem, the increase in hardware cost, and the inconvenient use in many applications.

Another approach is to use the coded exposure photography. Coded exposure photography was initially proposed by Raskar et al. for motion deblurring purpose.[5)] This kind of exposure control is now usually called flutter shutter photography in order to distinguish with the pixel-wise coded exposure techniques. Veeraraghavan et al. used a single flutter shutter camera to capture HSV of the periodic scenes which had a very sparse representation under the Fourier basis.[6)] Apparently, this scheme is very limited in practice since it can be solely applied for periodic signals. Holloway et al. also proposed to use a single flutter shutter camera to recover HSV.[7)] They depend on minimizing the total variation regularization along the temporal dimension, the temporal super-resolution ratio can only reach around 10×.

Recently, the pixel-wise coded exposure architecture has been proposed to perform HSV. Reddy et al. constructed a programmable pixel-wise compressive camera (P2C2) by employing a liquid crystal on silicon (LCoS) device and exploited the spatial redundancy using sparse representations and the temporal redundancy using brightness constancy.[8)] Hitomi et al.[9)] and Liu et al.[10)] described a high-speed imaging system with pixel-wise coded exposure and achieved the sparse representation of videos by learning an over-complete dictionary. Portz et al. used random pixel-wise exposure to reconstruct a high-speed HDR video from the coded input.[11)] The use of pixel-wise coded exposure leads to powerful results of capturing HSV with high compressions even for complex scenes. However, pixel-wise coded exposure requires advanced hardware such as liquid crystal on silicon or digital micro-mirror devices that would be difficult to fit into smaller cameras.[11)] Thus, the hardware implementation of the pixel-wise coded exposure is challenging and is a significant deviation from current commercial camera designs.[7)]

In consideration of easy achievement in modern sensors, we further exploit the possibility of using a single flutter shutter camera to achieve HSV reconstruction in this paper. In fact, flutter shutter photography has been already supported by several machine vision cameras. The closest prior work is proposed by Holloway et al.[7)] They model the temporal redundancy of videos using the total variation of videos along the temporal direction, without modeling the

*E-mail address: khhuang.nudt@gmail.com

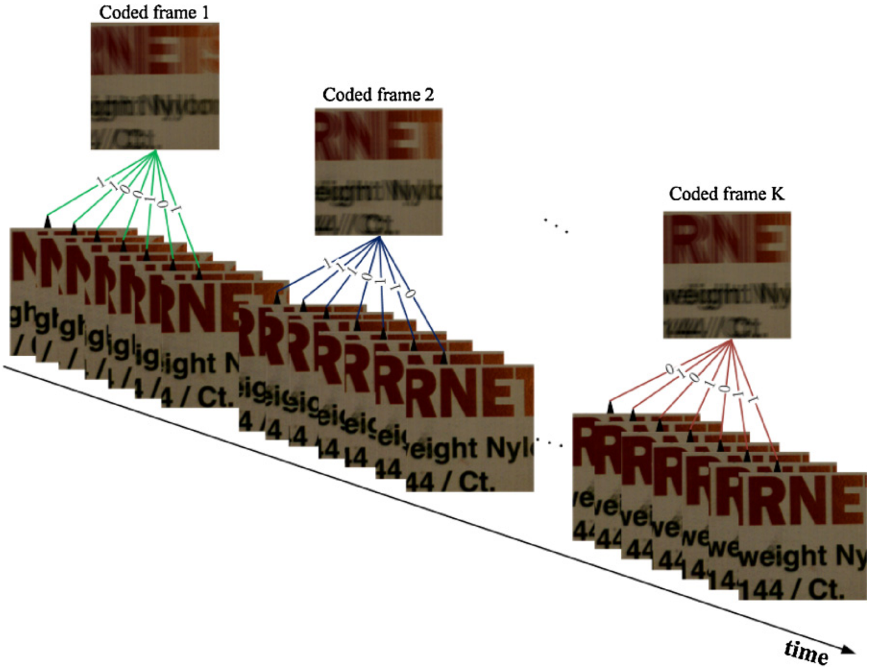

Fig. 1. (Color online) Coded sampling process of the proposed method.

spatial redundancy of videos. As a consequence, the reconstruction ability of their method is limited. This paper proposes to use a 3D hyperbolic wavelet basis based on Kronecker product[12,13] to jointly model the spatial and temporal redundancy of videos. Besides, we incorporate the total variation of temporal correlations in HSV as a prior knowledge to further enhance our reconstruction quality. The hyperbolic wavelet basis can simultaneously model all types of structure present on all the video dimensions with different scales, nicely catering for the nature of HSV that it has different degrees of smoothness along its spatial and temporal dimensions.

The idea of employing hyperbolic wavelets to exploit the spatial-temporal redundancy comes from Ref. 13. They concerned more about the compression of multidimensional signals, while we want to explore the possibility of reconstructing HSV with a low-speed flutter shutter camera. The remainder of the paper is organized as follows: Section 2 describes the coded sampling process of the proposed method. Section 3 depicts the 3D hyperbolic wavelet basis based on Kronecker product and presents our reconstruction model. The experimental results are reported in Sect. 4 and we conclude in Sect. 5.

2. Coded Sampling via Single Flutter Shutter Camera

A flutter shutter camera opens and closes the shutter according to a predefined binary pseudo random sequence within the exposure duration to modulate the incoming light. Here, we use the flutter shutter camera to obtain a coded video where each coded frame is a linear combination of the underlying high-speed frames along the temporal dimension. Figure 1 depicts the coded sampling process of the proposed method. If the underlying HSV $\mathbf{f}$ has N frames and each frame is a $m \times n$ two dimensional image denoted by $\mathbf{f}_t$, then through the coded sampling process, each coded frame $\mathbf{y}_k$ is given by an exposure of L high-speed frames with a binary random sequence $\mathbf{b}_k = (b_{k,1}, b_{k,2}, \ldots, b_{k,L}), k = 1, 2, \ldots, K$:

$$\mathbf{y}_k = \sum_{t=1}^{L} b_{k,t}\mathbf{f}_{(k-1)L+t} + \boldsymbol{\eta}_k, \quad k = 1, 2, \ldots, K, \tag{1}$$

where $\boldsymbol{\eta}_k \in \mathbb{R}^{m\times n}$ is the corresponding measurement noise, $\mathbf{b}_k \in \mathbb{R}^{1\times L}$, $\mathbf{y}_k \in \mathbb{R}^{m\times n}$ and the captured low-speed coded video $\mathbf{y} = (\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_K) \in \mathbb{R}^{m\times n\times K}$, $K = N/L$.

Let $\mathbf{y}(u, v)^{\mathrm{T}} = (\mathbf{y}_1(u, v), \mathbf{y}_2(u, v), \ldots, \mathbf{y}_K(u, v))^{\mathrm{T}}$ be the consecutive voxels of the observed coded video along the temporal dimension at spatial position (u, v), and $\mathbf{f}(u, v)^{\mathrm{T}} = (\mathbf{f}_1(u, v), \mathbf{f}_2(u, v), \ldots, \mathbf{f}_N(u, v))^{\mathrm{T}}$ be the corresponding voxel series of the underlying HSV at the same spatial position. The signal observation model along the temporal dimension can be written as

$$\mathbf{y}(u, v)^{\mathrm{T}} = \mathbf{B}_{K\times N}\mathbf{f}(u, v)^{\mathrm{T}} + \boldsymbol{\eta}(u, v)^{\mathrm{T}}, \tag{2}$$

where $\boldsymbol{\eta}(u, v)^{\mathrm{T}} = (\boldsymbol{\eta}_1(u, v), \boldsymbol{\eta}_2(u, v), \ldots, \boldsymbol{\eta}_K(u, v))^{\mathrm{T}}$ and $\mathbf{B}_{K\times N}$ is a block diagonal matrix made of K binary pseudo sequences as follows

$$\mathbf{B}_{K\times N} = \begin{bmatrix} \mathbf{b}_1 & \mathbf{0}_{1\times L} & \cdots & \mathbf{0}_{1\times L} \\ \mathbf{0}_{1\times L} & \mathbf{b}_2 & \cdots & \mathbf{0}_{1\times L} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{1\times L} & \mathbf{0}_{1\times L} & \cdots & \mathbf{b}_K \end{bmatrix}. \tag{3}$$

Let vec($\mathbf{y}$) be the vectorized format of the observed coded video $\mathbf{y}$ by stacking all the $m \times n \times K$ voxels into one column in temporal-vertical-horizontal order. And let vec($\mathbf{f}$) be the vectorized format of the desired HSV $\mathbf{f}$ in the same way. Then, we have

$$\mathrm{vec}(\mathbf{y}) = \boldsymbol{\Phi}\mathrm{vec}(\mathbf{f}) + \mathrm{vec}(\boldsymbol{\eta}), \tag{4}$$

where $\mathbf{\Phi} \in \mathbb{R}^{mnK \times mnN}$ is the measurement matrix which is made of matrix $\mathbf{B}_{K\times N}$ as

$$\mathbf{\Phi} = \begin{bmatrix} \mathbf{B}_{K\times N} & \mathbf{0}_{K\times N} & \cdots & \mathbf{0}_{K\times N} \\ \mathbf{0}_{K\times N} & \mathbf{B}_{K\times N} & \cdots & \mathbf{0}_{K\times N} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{K\times N} & \mathbf{0}_{K\times N} & \cdots & \mathbf{B}_{K\times N} \end{bmatrix}. \tag{5}$$

In Eq. (4), the number of unknown variables is much larger than the available equations. Therefore, to recover the underlying HSV $\mathbf{f}$ from the captured low-speed coded video $\mathbf{y}$ based on Eq. (4) is a severely under-determined problem, which has infinite number of solutions.

3. HSV Recovery

3.1 The unsymmetric structure of HSV

Fortunately, as the high-speed videos have significant spatial-temporal redundancy, inspired by the advances in compressive sensing, we decide to use the video priors to get a stable reconstruction by solving a convex optimization problem. The most common used video prior is that the underlying HSV is sparse or near sparse when represented in some appropriate transform basis, e.g., Wavelet, Fourier and DCT. In this paper, we focus on the wavelet transform which is widely used nowadays for sparse representations of natural images or videos. There are usually two ways to transform the video signal for getting a sparse representation. One is to apply a 2D wavelet basis to the video frame by frame and does not consider any temporal correlations. The other is to use a 3D isotropic wavelet basis to jointly model the spatial-temporal redundancy of the video signal. The sparse representation using a 3D isotropic wavelet basis is usually better than using a 2D wavelet basis frame by frame due to the incorporation of exploiting the temporal redundancy. However, the performance of the 3D isotropic wavelet basis is not optimal because the properties of the videos are not symmetric along the spatial and temporal dimensions, especially for HSV signals, which means that there are different degrees of smoothness along different dimensions in videos.

We selected the first 250 high-speed frames from "card_mons" dataset and "PendCar_lowres" dataset[14] respectively and selected the first 100 high-speed frames (cropped to 256 × 256) from "ResolutionChart" Dataset.[15] Table 1 describes the average standard deviations (STD) of three high-speed video sequences along horizontal, vertical and temporal dimensions, respectively. The result shows that the STD along the temporal dimension is much smaller than the STDs along the horizontal and vertical dimensions, while the STDs along the horizontal and vertical dimensions are very close. Therefore, it is reasonable to apply the transform along the temporal dimension in a different way from the transforms along the horizontal and vertical dimensions when using a 3D wavelet transform in the HSV processing.

Ideally, we should formulate a wavelet basis that can simultaneously account for all the structures present in the HSV. In fact, multidimensional signals often reveal structure in each mode which allows one to adopt good approximate representations based on the Kronecker product of dictionaries associated to each one of the modes.[12] Next we will show that the Kronecker product matrices offer a natural means of generating such a sparsity basis.

Table 1. Average STDs of HSV sequences along horizontal, vertical, and temporal dimensions.

	Horizontal	Vertical	Temporal
card_mons	64.797	69.894	15.726
PendCar_lowres	40.991	35.583	4.661
ResolutionChartDataset	28.733	28.962	11.043

3.2 Kronecker product for sparsity basis

Given two matrices $\mathbf{A} \in \mathbb{R}^{P\times Q}$ and $\mathbf{B} \in \mathbb{R}^{R\times S}$, the Kronecker product $\mathbf{A} \otimes \mathbf{B} \in \mathbb{R}^{PR\times QS}$ is defined by

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & \cdots & a_{1Q}\mathbf{B} \\ \vdots & \ddots & \vdots \\ a_{P1}\mathbf{B} & \cdots & a_{PQ}\mathbf{B} \end{bmatrix}. \tag{6}$$

For a 3D video signal $\mathbf{f} \in \mathbb{R}^{m\times n\times N}$, its mode-$d$ vectors are obtained by fixing every index except the one in mode-d, where $d \in \{1,2,3\}$. For example, the mode-3 vector of $\mathbf{f}$ is denoted as $\mathbf{f}_{i,j,\bullet} = [\mathbf{f}(i,j,1), \mathbf{f}(i,j,2), \ldots, \mathbf{f}(i,j,N)]$. The default vectorized format of a video is usually defined as stacking all its mode-1 vectors in one column. However, since we implement compressive sampling along the temporal dimension of the video, the vectorized format of a video in this paper is obtained by stacking all the mode-3 vectors in one column.

From the definition of the Kronecker product, it is easy to find that we can rewrite our measurement matrix in Eq. (5) as

$$\mathbf{\Phi} = \mathbf{I}_{mn} \otimes \mathbf{B}_{K\times N} = \mathbf{I}_{mn} \otimes [\mathrm{blkdiag}(\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_K)], \tag{7}$$

where $\mathbf{I}_{mn}$ denotes the $mn \times mn$ identity matrix and $\mathrm{blkdiag}(\bullet)$ denotes an operator which can construct a block diagonal matrix.

More generally, for an N-dimensional signal $\mathbf{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$, we assume each mode-$d$ vector is sparse or near sparse in the basis $\mathbf{\Psi}_d$. Then a Kronecker sparsity basis can be obtained by the Kronecker product as $\mathbf{\Psi} = \mathbf{\Psi}_N \otimes \cdots \otimes \mathbf{\Psi}_2 \otimes \mathbf{\Psi}_1$.[12] We can encode the vectorized format of $\mathbf{X}$ using a single transformation with this Kronecker sparsity basis, i.e.,

$$\mathrm{vec}(\mathbf{X}) = \mathbf{\Psi}^{\mathrm{T}}\mathrm{vec}(\boldsymbol{\theta}) = (\mathbf{\Psi}_N \otimes \cdots \otimes \mathbf{\Psi}_2 \otimes \mathbf{\Psi}_1)^{\mathrm{T}}\mathrm{vec}(\boldsymbol{\theta}), \tag{8}$$

where $\mathrm{vec}(\boldsymbol{\theta})$ is the vectorized format of coefficients $\boldsymbol{\theta}$.

For our HSV signal processing, we have

$$\mathrm{vec}(\mathbf{f}) = (\mathbf{\Psi}_h \otimes \mathbf{\Psi}_v \otimes \mathbf{\Psi}_t)^{\mathrm{T}}\mathrm{vec}(\boldsymbol{\theta}), \tag{9}$$

where $\mathbf{\Psi}_h$, $\mathbf{\Psi}_v$, and $\mathbf{\Psi}_t$ are 1D transform bases for the horizontal dimension, the vertical dimension and the temporal dimension, respectively.

3.3 Hyperbolic wavelets

Video signals can be represented by wavelets in a more compact way because the discontinuities in video usually take substantially fewer wavelet basis functions than sine-cosine basis functions (e.g., DCT, DFT) to achieve a comparable approximation. The wavelet decomposition of a 1D signal $f(t), t \in [0,1]$ of size 2^n is given by

$$f(t) = c_{00}\phi(t) + \sum_{j=0}^{n-1}\sum_{k=0}^{2^j-1} d_{jk}\psi_{jk}(t), \tag{10}$$

where $\phi(t)$ is the scaling function and $\psi_{jk}(t)$ is the wavelet function at scale j and position k. The scaling coefficient c_{00} and the wavelet coefficients d_{jk} at scale j and position k compose the final wavelet transform coefficients; the support of the wavelet function ψ_{jk} at scale j and position k is about $[k2^{-j}, (k+1)2^{-j}]$. If we write Eq. (10) in matrix-vector form as our earlier notation as $\mathbf{x} = \mathbf{\Psi}\boldsymbol{\theta}$, then $\mathbf{\Psi}$ is a matrix with the scaling and wavelet functions of scales $1, 2, \ldots, n$ as columns and $\boldsymbol{\theta}$ is a vector containing the scaling and wavelet coefficients with a form as $\boldsymbol{\theta} = [c_{00}, \psi_{00}, \psi_{10}, \psi_{11}, \psi_{20}, \ldots]^{\mathrm{T}}$.

From Table 1 we know that the STDs along the horizontal and vertical dimensions are very close, which means the popular 2D isotropic wavelet will suffice for an image transform. But for videos, especially the high-speed videos which have more similarity along the temporal dimension, the 3D isotropic wavelet transform with the same parameter of scale in all three dimensions is usually impotent. Hence, regularization of 3D wavelet transform coefficients to solve the under-determined system in Eq. (4) usually results in poor reconstruction quality. Some researchers try to sparsify a video across the temporal dimension against the motion information,[8,14] such as optical flow. The main issue, in this case, is the motion information is not available before acquisition, and an iterative and computationally demanding estimation procedure should be carried out.

To overcome the above challenges, we propose to use the 3D hyperbolic wavelet transform to simultaneously exploit the redundancy in HSV along all three dimensions. A 3D hyperbolic wavelet basis can be simply defined as Kronecker product of three 1D wavelet bases, i.e.,

$$\boldsymbol{\psi}_{j_1,j_2,j_3,k_1,k_2,k_3} = \boldsymbol{\psi}_{j_3,k_3} \otimes \boldsymbol{\psi}_{j_2,k_2} \otimes \boldsymbol{\psi}_{j_1,k_1}, \tag{11}$$

where $(j_1, j_2, j_3) \in \mathbb{N}^{*3}$ and $(k_1, k_2, k_3) \in \mathbb{Z}^3$. A 3D hyperbolic wavelet basis is obtained from Kronecker product of all possible combinations of three 1D wavelet bases with different scales, while a 3D isotropic wavelet basis makes use of the same scale along all dimensions, i.e., in this case, $j_1 = j_2 = j_3$.

3.4 Reconstruction model incorporating TV regularization

Zhang[16] demonstrated the flexibility to incorporate total variation (TV) prior information into L1-minimization decoding models and gave a theoretical guarantee that adding prior information can never hurt but possibly enhance recoverability. For HSV, there is more temporal redundancy than spatial redundancy because the exposure interval of each frame is very small. In this case, the total variation of the temporal correlations in HSV should be very small, which is defined as

$$\|\mathbf{f}\|_{TV} = \|(\mathbf{I}_{mn} \otimes \nabla_t)\mathrm{vec}(\mathbf{f})\|_1, \tag{12}$$

where ∇_t is the first-order differential operator along the temporal dimension, so that

$$\nabla_t = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ -1 & 1 & 0 & \ddots & \vdots \\ 0 & -1 & 1 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & -1 & 1 \end{bmatrix}. \tag{13}$$

Combining the ℓ_1 regularization based on the 3D hyperbolic wavelets and the temporal TV regularization, we propose to solve the under-determined problem in Eq. (4) by solving the following ℓ_1 minimization problem

$$\arg\min_{\mathbf{f}} \|(\boldsymbol{\psi}_h \otimes \boldsymbol{\psi}_v \otimes \boldsymbol{\psi}_t)\mathrm{vec}(\mathbf{f})\|_1 + \mu\|\mathbf{f}\|_{TV}$$
$$\text{s.t. } \|\mathbf{\Phi}\mathrm{vec}(\mathbf{f}) - \mathrm{vec}(\mathbf{y})\|_2 \le \sigma, \tag{14}$$

where σ is the variance of the measurement error. In all our experiments we fix the parameters $\mu = 0.5$ and $\sigma = 0.1$.

4. Experimental Results

4.1 Validity of the proposed method

We first verified the performance and capability of the proposed method through simulation. We simulated the low-speed coded video using Eq. (1) with a 1000 fps video sequence "ResolutionChart" dataset credited to Amit Agrawal. For implementing convenience, we cropped the video sequence to a spatial resolution of 256 × 256 pixels and normalized the pixel values to the range of $[0, 255]$. Figures 2(b) and 2(c) are the reconstructed high-speed frames (frame 56) using our proposed method and Holloway's method[7] at 8× temporal super-resolution, respectively. Both of the two results have good visual quality and high peak signal-to-noise ratio (PSNR). Figures 2(d) and 2(e) are the corresponding recovered high-speed frames using the proposed method and Holloway's method at 16× temporal super-resolution, respectively. Although there is abundant texture information at the central part of the frame, the quality of our reconstructed result is good enough to distinguish these details, and the improvement of the visual quality is apparent over the result in Fig. 2(e). Because Holloway's method only exploits the redundancy along the temporal dimension, the quality of the reconstructed video decays quickly when the temporal super-resolution factor increases, while our method can still maintain good reconstructed quality at 16× temporal super-resolution. The fine details in Fig. 2(f) are the close-up versions of the regions in Figs. 2(d) and 2(e) labeled with rectangles, respectively. The comparison of the close-up versions further verifies the superiority of the proposed method.

Figure 3 depicts the PSNR curves of the first 64 recovered high-speed frames of "ResolutionChart" dataset using the proposed method and Holloway's method at 16× temporal super-resolution and 8× temporal super-resolution, respec-

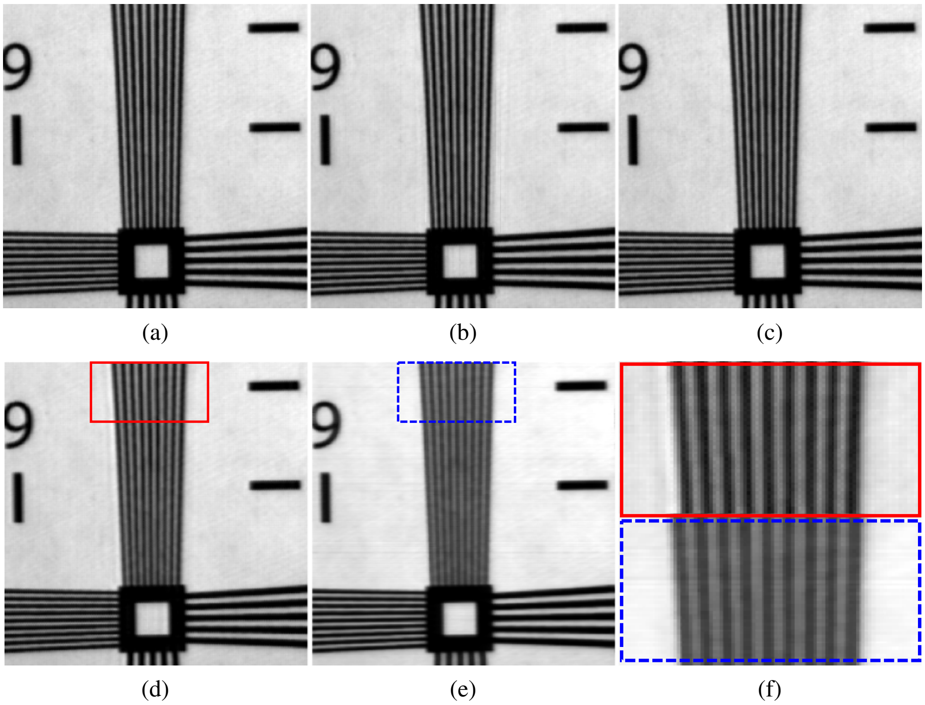

Fig. 2. (Color online) Reconstructed results of HSV sequence "ResolutionChart" dataset credited to Amit Agrawal. (a) The ground truth. (b) and (c) are the reconstructed frames (frame 56) using our method and Holloway's method at 8× temporal super-resolution with PSNR of 33.03 and 32.59 dB, respectively. (d) and (e) are the corresponding reconstructed frames using our method and Holloway's method at 16× temporal super-resolution with PSNR of 29.18 and 22.56 dB, respectively. (f) is the close-up version of the corresponding regions labeled with rectangles in (d) and (e).

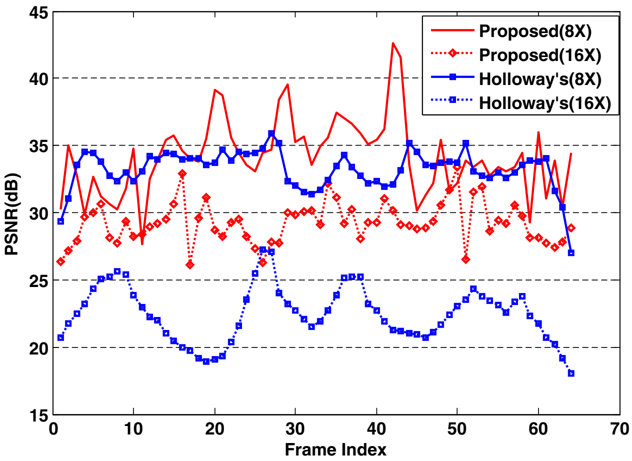

Fig. 3. (Color online) PSNR values of the first 64 recovered frames using the proposed method and Holloway's method at 8× and 16× temporal super-resolution respectively with respect to frame index.

tively. The PSNR values obtained by the proposed method and Holloway's method at 8× temporal super-resolution fluctuate around 34 and 33 dB respectively, while the PSNR values derived by the proposed method and Holloway's method at 16× temporal super-resolution fluctuate around 29 and 22.5 dB, respectively. Apparently, with the increase of temporal super-resolution factor, the decay speed of the quality of reconstructed HSV of our proposed method is not as quick as Holloway's method in terms of PSNR.

4.2 *The influence of sparse representations and TV regularization*

In this section, one intent is to demonstrate the superiority of the proposed use of 3D hyperbolic wavelets by showing the reconstruction performance comparison for different sparse representations, and the other is to show the improvement of incorporating TV prior knowledge in the proposed method. The input low-speed coded frames were simulated on "Cardmons" dataset using Eq. (1). "Cardmons" dataset has a frame rate of 250 fps and the spatial resolution of each frame is 256×256. For a fair comparison, we only use the sparse representations to exploit the video redundancy without adding the TV prior knowledge in the reconstruction process, i.e., we set $\mu = 0$ in Eq. (14). Figures 4(b)–4(e) are the recovered frames (frame 11) at 16× temporal super-resolution using 3D DFT, 3D DCT, 3D isotropic wavelets and 3D hyperbolic wavelets for sparse representations in the reconstruction process, respectively. It can be seen visually that the result in Fig. 4(e) gets the best performance, which demonstrates the superiority of using 3D hyperbolic wavelets for sparse representation of HSV. Since we did not incorporate the TV prior knowledge, the reconstructed result in Fig. 4(e) is actually not ideal; there are some ghost artifacts around the edges in the frame, especially in the playing card area. Hence, the PSNR of Fig. 4(e) only achieves 22.40 dB.

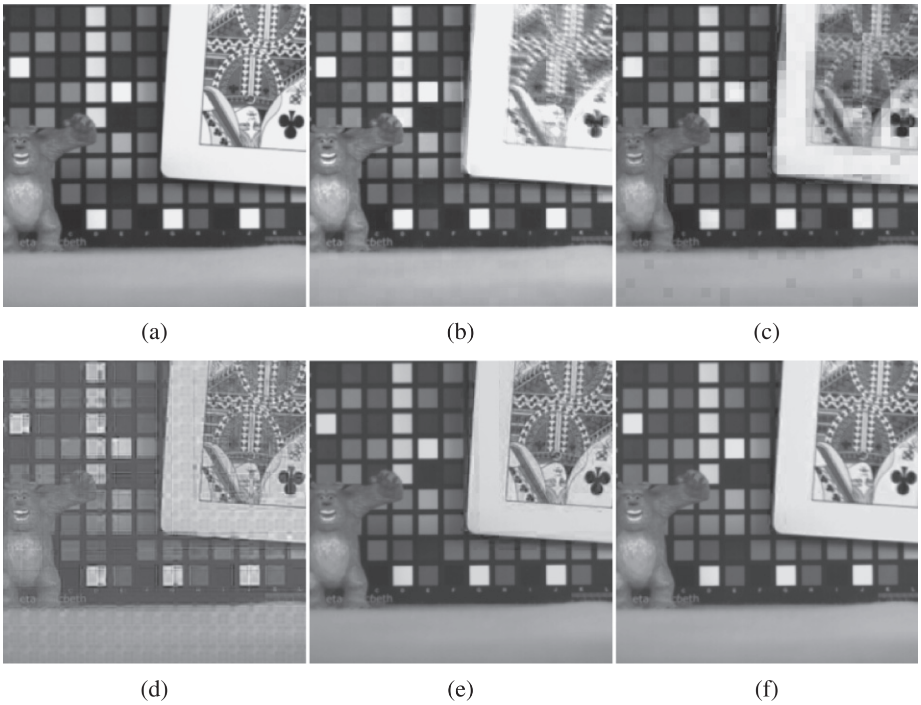

Fig. 4. Performance comparison of different sparse representations. (a) The ground truth. (b)–(e) are the recovered high-speed frames (frame 11) at 16× temporal super-resolution using 3D DFT, 3D DCT, 3D isotropic wavelets and 3D hyperbolic wavelets for sparse representations in the reconstruction process with PSNR of 20.24, 19.53, 17.34, and 22.40 dB, respectively. (f) is the recovered frame using the proposed method which incorporates TV prior knowledge, with PSNR of 27.49 dB.

Figure 4(f) is the final result recovered by the proposed method using 3D hyperbolic wavelets and incorporating the TV prior knowledge at 16× temporal super-resolution. It is apparent that the ghost artifacts are significantly reduced. Actually, the PSNR of Fig. 4(f) achieves 27.49 dB which is about 5 dB higher than the PSNR of Fig. 4(e) that did not incorporates the TV prior knowledge in the reconstruction process.

4.3 Real-data results

We implemented the HSV system by employing a Point Grey Flea2 video camera to capture a moving toy train. Flea2 works in IEEE DCAM Trigger mode 5 which supports coded exposure functionality. The coded exposure pattern was provided by an external trigger using an Arduino Duemilanove board. The experimental setup is given in Fig. 5.

Flea2 video camera has a maximum frame rate of 7.5 fps in trigger mode 5 due to hardware limitation. In our experiment, the random binary sequences were set to 16 in length, resulted a compression factor of 16×. Hence, the HSV we recovered using the proposed method had a frame rate of 120 fps. Figure 6 (top-left) shows one of the coded frames captured by our camera. For simplicity and without loss of generality, we selected fixed regions which had abundant texture information from these coded frames as the input of the proposed method, as shown in Fig. 6 (top-right). The bottom row in Fig. 6 shows three reconstructed frames. Notice that the blurring is significantly reduced, which demonstrates the validity of the proposed method.

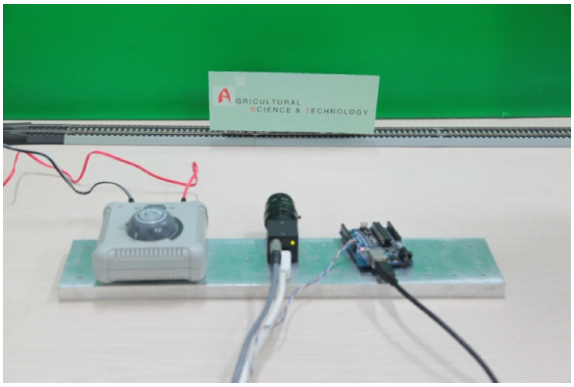

Fig. 5. (Color online) Experimental setup.

4.4 Implementation details

In our experiments, the 3D wavelet basis is built with order 8 Daubechies wavelet bases deployed by the Rice Wavelet Toolbox (RWT).[17] In consideration of the computational complexity, we divided the video into 16 × 16 patches and reconstructed them in sequence. Recalling our coded sampling process, we know that the measurement matrix $\mathbf{\Phi}$ is a block diagonal matrix with many zeros which means the randomness of $\mathbf{\Phi}$ will decrease with the number of input coded frames. In addition to computational load, we cannot recover arbitrarily many high-speed frames at a time. We tested the input coded frame number K in our experiment, and found $K = 4$ was a good choice.

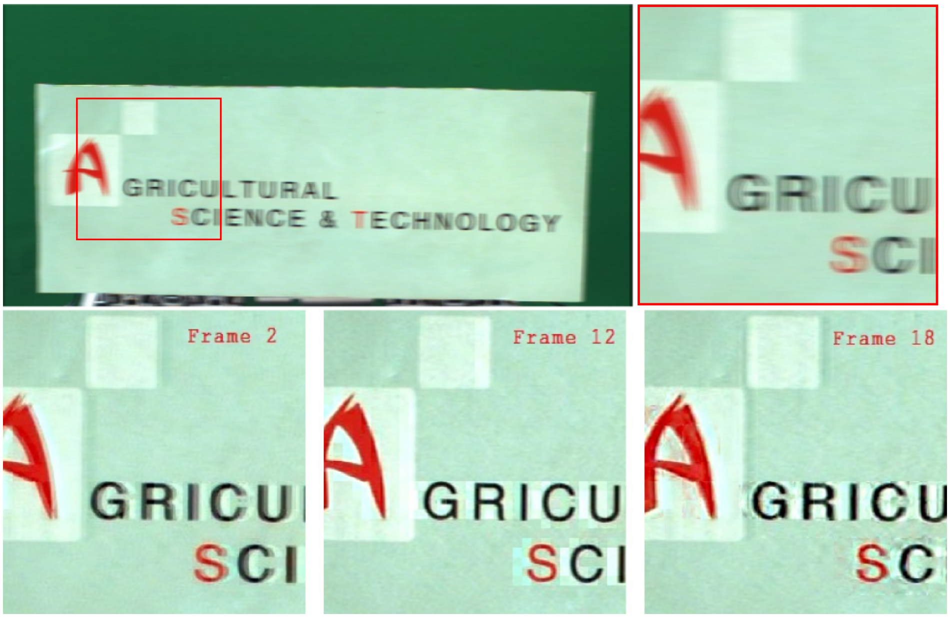

Fig. 6. (Color online) Real-data results. Top-left: one of the captured coded frames; top-right: selected region as input of the proposed method. Bottom row: three reconstructed high-speed frames (frame 2, frame 12, and frame 18).

5. Conclusion

In consideration of easy implementation in modern sensors, this paper further exploits the possibility of the recovery of HSV using a single flutter shutter camera. We fully account on the nature of HSV that it has different degrees of smoothness along different dimensions and construct a 3D hyperbolic wavelet basis based on Kronecker product to jointly model its spatial and temporal redundancy. We also utilize the flexibility of ℓ_1 minimization problem and incorporate TV regularization in our reconstruction model. Experimental results on simulated video and real data are promising and demonstrate the efficacy of the proposed method. CMOS sensor cameras usually use rolling shutter and can be controlled very easily. Combination of rolling shutter and coded exposure together to find some interesting applications will be our future work.

Acknowledgments

The work of K. H. Hunag is supported by China Scholarship Council during his visit to McMaster University, and in part by the National Natural Science Foundation of China under Grants No. 60902093 and No. 61170159. K. H. Huang thanks McMaster University, ECE Department, for hosting him and thanks Professor Xiaolin Wu for his unconditional assistance and his sagacious suggestions.

References

1) X. Wu and R. Pournaghi: Proc. 17th IEEE Int. Conf. Image Processing, 2010, p. 577.
2) B. Wilburn, N. Joshi, V. Vaish, E. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy: ACM SIGGRAPH, 2005, p. 765.
3) E. Shechtman, Y. Caspi, and M. Irani: IEEE Trans. Pattern Anal. Mach. Intell. **27** (2005) 531.
4) A. Agrawal, M. Gupta, A. Veeraraghavan, and S. G. Narasimhan: Proc. 23rd IEEE Int. Conf. CVPR, 2010, p. 599.
5) R. Raskar, A. Agrawal, and J. Tumblin: ACM SIGGRAPH, 2006, p. 795.
6) A. Veeraraghavan, D. Reddy, and R. Raskar: IEEE Trans. Pattern Anal. Mach. Intell. **33** (2011) 671.
7) J. Holloway, A. C. Sankaranarayanan, A. Veeraraghavan, and S. Tambe: Proc. IEEE Int. Conf. Computational Photography, 2012, p. 1.
8) D. Reddy, A. Veeraraghavan, and R. Chellappa: Proc. IEEE Int. Conf. CVPR, 2011, p. 329.
9) Y. Hitomi, J. Gu, M. Gupta, T. Mitsunaga, and S. K. Nayar: Proc. IEEE Int. Conf. Computer Vision, 2011, p. 287.
10) D. Liu, J. Gu, Y. Hitomi, M. Gupta, T. Mitsunaga, and S. K. Nayar: IEEE Trans. Pattern Anal. Mach. Intell. **36** (2014) 248.
11) T. Portz, L. Zhang, and H. Jiang: Proc. IEEE Int. Conf. Computational Photography, 2013, p. 1.
12) C. F. Caiafa and A. Cichocki: Neural Comput. **25** (2013) 186.
13) M. F. Duarte and R. G. Baraniuk: IEEE Trans. Image Process. **21** (2012) 494.
14) A. C. Sankaranarayanan, C. Studer, and R. G. Baraniuk: Proc. IEEE Int. Conf. Computational Photography, 2012, p. 1.
15) A. Agrawal and R. Raskar: Proc. IEEE Int. Conf. CVPR, 2009, p. 2560.
16) Y. Zhang: Rice University Tech. Rep. 08-11 (2008).
17) R. Baraniuk, H. Choi, R. Neelamani, V. Ribeiro, J. Romberg, H. Guo, F. Fernandes, B. Hendricks, R. Gopinath, M. Lang, J. E. Odegard, and D. Wei: Rice Wavelet Toolbox (RWT), [http://dsp.rice.edu/software/rice-wavelet-toolbox].