



Optics Letters

Near-infrared temporal compressive imaging for video

QUN ZHOU,^{1,2} JUN KE,^{1,2,*} AND EDMUND Y. LAM³

¹School of Optics and Photonics, Beijing Institute of Technology, Beijing 100081, China

²Key Laboratory of Photoelectronic Imaging Technology and System, Ministry of Education of China, Beijing 100081, China

³Department of Electrical and Electronic Engineering, University of Hong Kong, Pokfulam, Hong Kong, China

*Corresponding author: jke@bit.edu.cn

Received 12 December 2018; revised 9 February 2019; accepted 25 February 2019; posted 4 March 2019 (Doc. ID 354791); published 25 March 2019

Without decreasing spatial resolution, temporal compressive imaging (TCI) can improve the temporal resolution of an imaging sensor and relax the requirement of the data readout speed in high-speed imaging. In this Letter, we describe a near-infrared TCI system that can reconstruct 500 fps videos from coded measurement frames sampled at 50 fps. © 2019 Optical Society of America

<https://doi.org/10.1364/OL.44.001702>

Compared to visible light, near-infrared (NIR) band light has better penetration through scattering media and is more tolerant to temperature variations. Therefore, NIR imaging is often used to acquire important environmental features that visible light cannot detect. In NIR imaging, the spatial resolution (SR) and temporal resolution (TR) of an imaging sensor are restricted by current industrial manufacturing and technology level. In general, there is a tradeoff between SR and TR. A camera with high SR, but low TR, often results in motion blur. On the other hand, if a camera's TR or its frame rate increases, then its SR drops. For example, a camera in a visual band can have a frame rate exceeding 10,000 fps, but with small SR such as 32×32 . Building a high-speed and high SR camera is still an important topic in research and industry. In particular, in an infrared band, the frame rate of a high-speed camera is nowhere near the speed of a high-speed visual band camera. Furthermore, high cost is also a major issue for infrared high-speed camera. Thus, in this Letter, we study a new technology, temporal compressive imaging (TCI), for high-speed cameras in an IR band.

TCI uses a low frame rate camera to collect a temporally modulated and compressive video, and then reconstruct the original high frame rate video from its measurements. In addition to producing a high-frame-rate video using a low-speed response camera without sacrificing high SR, TCI relaxes imaging sensors' requirement to a large bandwidth but keeping the same light sensitivity. In past several years, several groups did simulation and experimental works in this field. In 2011, Reddy *et al.* proposed a programmable pixel compressive camera (P2C2) [1] that can produce temporal super-resolution

videos at 200 fps using a 25 fps camera with a liquid crystal on silicon modulator. In 2013, Llull *et al.* demonstrated coded aperture compressive temporal imaging [2] with experimental hardware. Multiple reconstruction algorithms, including optical flow [1], over-complete dictionary learning [3], two-step iterative shrinkage thresholding (TwIST) [4], generalized alternating projection [5], and Gaussian mixture model (GMM) [6] are used to recover high-speed video from temporal compressive measurements. In 2014, Yang *et al.* developed an online-updated GMM algorithm to improve reconstruction quality [6]. Later, they developed a method [7] for adaptive temporal compression ratio. In 2017, Yuan *et al.* built a system setup [8] with superior performance to recover high-speed videos using double optical paths. Recently, deep learning is also applied to temporal video CS [9]. It demonstrates better reconstruction quality using less reconstruction time.

In this Letter, we describe a TCI setup for NIR imaging. We present the conventional imaging and TCI system diagrams in Figs. 1(a) and 1(b), respectively. In the conventional system, a blurred picture will be captured, as shown in Fig. 1(a), due to the slow speed of a camera compared to fast varying scenes. In the TCI system, during the exposure time of one frame, a spatial light modulator (SLM) varies T times sequentially to encode the original high-speed scene. Then from a single compressed frame, T frames can be reconstructed. Note that T is the temporal compression ratio. Therefore, the frame rate of the reconstructed video will be increased T times from the camera's frame rate.

The mathematical model of TCI can be expressed as

$$\mathbf{y} = \Phi \mathbf{x} + \mathbf{n}, \quad (1)$$

where $\mathbf{x} \in \mathbb{R}^{N_x N_y T \times 1}$ is a vector for T original high-speed video frames with size of $N_x \times N_y$. As discussed before, during one camera frame exposure time, T masks, $\{\mathbf{H}_i\}, i = 1, 2, \dots, T$ are used. If we rewrite \mathbf{H}_i into a diagonal matrix as $\text{diag}(\mathbf{H}_i)$, then the measurement matrix can be defined as $\Phi = [\text{diag}(\mathbf{H}_1), \text{diag}(\mathbf{H}_2), \dots, \text{diag}(\mathbf{H}_T)]$, while Φ has dimensions $(N_x * N_y) \times (N_x * N_y * T)$. Note that the number of rows in Φ is much less than the number of columns. $\mathbf{y} \in \mathbb{R}^{N_x N_y \times 1}$ is the measurement vector representing one frame measurement obtained by compressing T original frames, and $\mathbf{n} \in \mathbb{R}^{N_x N_y \times 1}$ is the noise

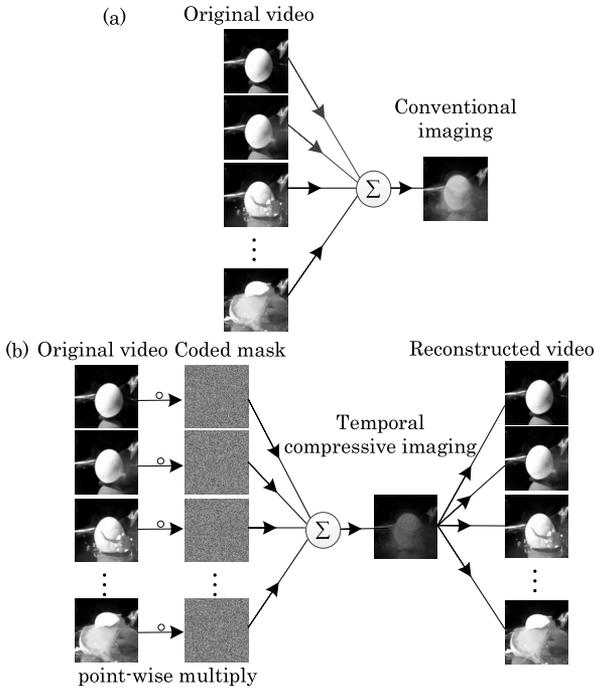


Fig. 1. (a) Conventional imaging and (b) TCI.

vector. With this model, TCI reconstruction becomes recovering \mathbf{x} given \mathbf{y} , Φ , and \mathbf{n} . In this Letter, we use two algorithms, TwIST and GMM, which are well known in compressive imaging society.

In the TwIST algorithm, we solve the following convex minimization problem for reconstruction:

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{y} - \Phi \mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_{TV}, \quad (2)$$

where $\lambda \in [0, +\infty)$ is a penalty parameter. In the regularization term, we use 2D total variation (2D-TV), since it performs better than a 3D format in simulation. The two-step iteration of the TwIST algorithm is as follows:

$$\begin{cases} \mathbf{x}_1 = \Gamma_\lambda(\mathbf{x}_0) \\ \mathbf{x}_{t+1} = (1 - \alpha)\mathbf{x}_{t-1} + (\alpha - \beta)\mathbf{x}_t + \beta\Gamma_\lambda(\mathbf{x}_t) \end{cases}, \quad (3)$$

where α, β are the parameters, which can affect the converging speed of TwIST. Their values can be set by some computational process, according to Ref. [4]. The denoising operation $\Gamma_\lambda(\mathbf{x}_t)$ can be determined by solving Eq. (4) using the Chambolle's projection algorithms [10]:

$$\min_{\mathbf{x}_t} \frac{1}{2} \|\mathbf{x}_t - \mathbf{x}_t^*\|_2^2 + \lambda \|\mathbf{x}_t\|_{TV}, \quad (4)$$

$$\mathbf{x}_t^* = \mathbf{x}_{t-1} + \Phi^T(\mathbf{y} - \Phi \mathbf{x}_{t-1}). \quad (5)$$

To use TwIST for reconstruction, we study three initialization conditions, TwIST-zero, TwIST-online, and TwIST-wiener. In TwIST-zero, we set the initial \mathbf{x} as a zero vector. TwIST-online applies the previously reconstructed frames as an initial value to reconstruct current frames. In TwIST-wiener, we use the preliminary reconstruction obtained using Wiener algorithm [11,12] as the initial value. $\mathbf{W} = \mathbf{R}_x \Phi^T (\Phi \mathbf{R}_x \Phi^T + \sigma^2 \mathbf{I})^{-1}$ is defined as Wiener operator, where \mathbf{R}_x is the object auto-correlation matrix, and Φ is a measurement matrix as previously

mentioned. \mathbf{I} is an identity matrix, and σ^2 is the variance of Gaussian noise. The second term considers the effect of noise for reconstruction. Using Wiener operator \mathbf{W} , the initial value is computed as $\mathbf{x}_{\text{est}} = \mathbf{W} \mathbf{y}$. Note that, due to the limited number of objects and the heavy computation for matrix \mathbf{R}_x , the object is divided into blocks to calculate the initial \mathbf{x} . We found that TwIST-wiener performs the best among the three initialized methods.

The second video reconstruction algorithm is based on the GMM, which assumes that an image signal is Gaussian distributed and can be described using the following prior distribution:

$$p(\mathbf{x}) = \sum_{k=1}^K \lambda_k N(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Lambda}_k), \quad (6)$$

where K denotes the number of mixture components. Matrices λ_k , $\boldsymbol{\mu}_k$, and $\boldsymbol{\Lambda}_k$ represent the weight, mean, and covariance of the k th mixture component, respectively. $N(\mathbf{n} | 0, \mathbf{R})$ is the probability distribution function of a Gaussian noise. According to Bayes theorem, the reconstructed object signal $\tilde{\mathbf{x}}$ can be obtained by Eqs. (7)–(9). More details about the derivation of these equations can be found in Ref. [6]. Parameters λ_k , $\boldsymbol{\mu}_k$, and $\boldsymbol{\Lambda}_k$ can be obtained by training other high-speed videos:

$$\boldsymbol{\omega}_k = \frac{\lambda_k N(\mathbf{y} | \Phi \boldsymbol{\mu}_k, \mathbf{R}^{-1} + \Phi \boldsymbol{\Lambda}_k \Phi^T)}{\sum_{l=1}^K \lambda_l N(\mathbf{y} | \Phi \boldsymbol{\mu}_l, \mathbf{R}^{-1} + \Phi \boldsymbol{\Lambda}_l \Phi^T)}, \quad (7)$$

$$\boldsymbol{\theta}_k = \boldsymbol{\Lambda}_k \Phi^T (\mathbf{R} + \Phi \boldsymbol{\Lambda}_k \Phi^T)^{-1} (\mathbf{y} - \Phi \boldsymbol{\mu}_k) + \boldsymbol{\mu}_k, \quad (8)$$

$$\tilde{\mathbf{x}} = \sum_{k=1}^K \boldsymbol{\omega}_k \boldsymbol{\theta}_k. \quad (9)$$

For the GMM model, we also discuss three algorithms for reconstruction. If the sample set for training λ_k , $\boldsymbol{\mu}_k$, and $\boldsymbol{\Lambda}_k$ repeatedly includes the latest reconstructed frames and excludes the earliest reconstructions as shown in Fig. 2, we name the algorithm as GMM-online algorithm [6]. Otherwise, we refer it as GMM-offline algorithm. In Fig. 2, the difference between the two is whether having the dotted line or not. To some extent, the online algorithm can reduce the dependence on training samples and make full use of reconstructed data. The third algorithm is GMM-wiener. In this algorithm, the training sample set is composed of the reconstructions obtained using a Wiener operator. λ_k , $\boldsymbol{\mu}_k$, and $\boldsymbol{\Lambda}_k$ are trained using this set. Then the GMM-offline algorithm is applied. It is apparent that the GMM-wiener algorithm is fairly quick speed, compared to

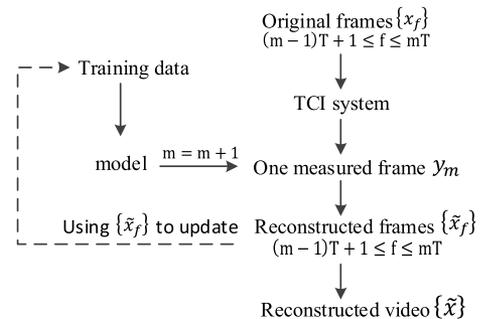


Fig. 2. Online-updated GMM algorithm.

the GMM-online method, because it avoids training model repeatedly.

To demonstrate TCI, we did simulated and experimental works. In simulation, a video for a bullet ejecting from a pistol in an IR band (Video I) and a video for a fast moving billiard in a visible band (Video II) are used. The compressive ratios are 8 in both cases. Because objects have high moving speed and a digital micromirror device (DMD) must vary no slower than an object, we choose random binary matrices to encode the original video. Each element in these matrices satisfies Bernoulli distribution and takes value 0 or 1 randomly with a probability 0.5. A Gaussian noise $N(0, 10^{-6})$ is added to simulate the real scene.

Six algorithms—TwIST-zero, TwIST-online, TwIST-wiener, GMM-offline, GMM-online, and GMM-wiener—are studied. In the TwIST-wiener, we divide each measurement frame into several 8×8 blocks and use Wiener to calculate the initial reconstructed $8 \times 8 \times 8$ video blocks. Then each frame is reconstructed as a whole unit using the algorithm defined in Eqs. (3)–(5). For GMM algorithms, because of heavy computation cost and requirement to large number of training samples, we divide a measurement frame into small 8×8 blocks, and process each block individually. The training data consist of several video clips which are unrelated to our testing videos. The number of Gaussian kernels, K , is chosen as 10 s.

In Fig. 3, we plot peak signal-to-noise ratio (PSNR) of reconstruction results for Videos I and II as functions of frames. Every set of reconstructed frames by one measurement frame has a periodic change. The boundary frames perform worst among each set, because they are short of adjacent frame information comparably. Joint reconstructing two sets of frames

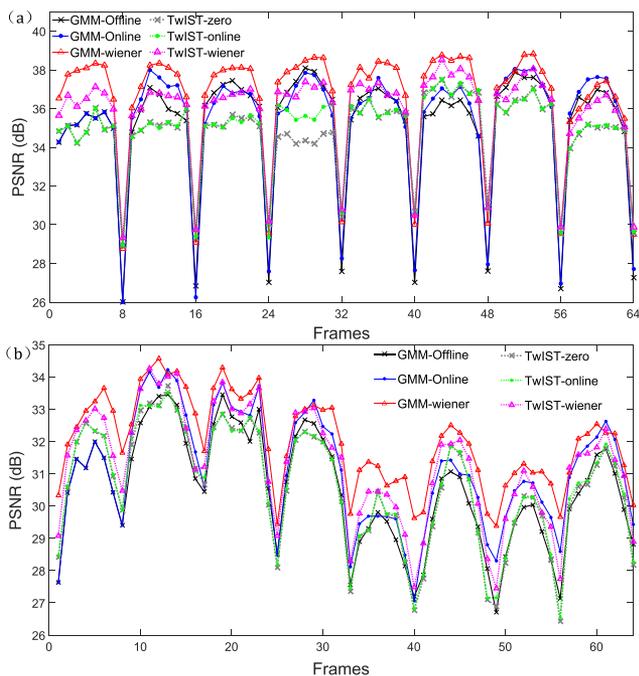


Fig. 3. (a) and (b) are the reconstructed results of video I (see Visualization 1, Visualization 2, Visualization 3, Visualization 4, Visualization 5, and Visualization 6) and II (see Visualization 7, Visualization 8, Visualization 9, Visualization 10, Visualization 11, and Visualization 12), respectively.

Table 1. Average PSNRs of Different Algorithms on Two Videos

PSNR (dB)	GMM-Offline	GMM-Online	GMM-Wiener	TwIST-Zero	TwIST-Online	TwIST-Wiener
Video I	35.15	35.33	36.61	34.81	34.94	35.88
Video II	30.51	31.09	32.00	30.58	30.63	31.29

from two measurements, or increasing the compressive ratio will mitigate the issue. Overall, it can be found that, TwIST-wiener and GMM-wiener have the best performance among the TwIST and the GMM algorithms. Additionally, the computational time for GMM-online is heavy, because it needs to train the GMM parameters repeatedly. Therefore, GMM-wiener runs fast. Table 1 shows the average PSNR results of different algorithms on Video I and II. It can be seen that GMM-wiener performs the best among the six algorithms.

In the experimental works, we use GMM-wiener and TwIST-wiener algorithms for NIR band video reconstruction first. Figure 4 presents our experimental setup with the real object, which is a self-made rotating target. One target board printed numbers uniformly is driven by an electronic motor for high-speed revolving around its central axis. Using this setup, infrared light from a dynamic scene, which is illuminated by a NIR illuminator, is imaged onto a programmable DMD via objective lens. The DMD works as a SLM to encode the image of the scene at high speed. Then the encoded light is accumulated and refocused onto a CCD array via an imaging lens. A NIR filter is inserted in front of the CCD array for filtering visible light. The working wave band in the system is 800–900 nm. The outputs of the array are low frame rate measurements. Notice that, to use TwIST-wiener and GMM-wiener, we need original high-speed videos to estimate \mathbf{R}_x and the initial GMM model. It is unrealistic to obtain high-speed videos using low speed sensors. To deal with the issue, we shorten the exposure time of our CCD such that clear instead of motion blurred images are captured in a low frame rate 50 fps. In our experiment, the compressive ratio is 10, and the noise \mathbf{n} is estimated as $N(0, 10^{-6})$.

The first experiment is in the NIR band. Figure 5(a) presents a scene in a NIR band with motion blur coming from a low frame rate 50 fps of CCD camera. Figure 5(b) shows a raw NIR measurement frame collected using the same camera. We can see random pattern on the scene. Figures 5(c)–5(d) show the reconstructed frames using GMM-wiener and TwIST-wiener algorithms, respectively. Compared to Fig. 5(a), the results are sharper and, thus, can provide more information of the high-speed moving object. Between the two algorithms, it also can be observed that GMM-wiener performs better

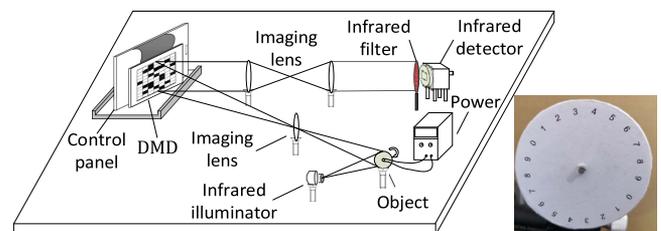


Fig. 4. Setup of TCI in a NIR waveband with the rotating object.

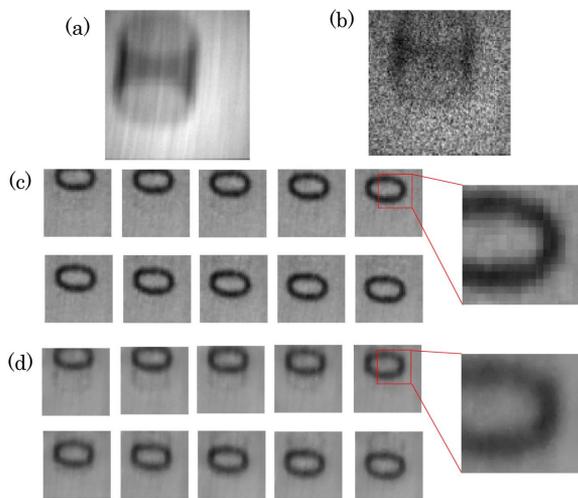


Fig. 5. Experimental results in a NIR. (a) One frame captured by a low frame rate camera; (b) one coded measurement frame; reconstructions using (c) GMM-wiener and (d) TwIST-wiener. The complete results can be found in [Visualization 13](#).

than TwIST-wiener for the rotating object. The results of GMM-wiener have clear edges and provide more details. However, from the right columns in Figs. 5(c)–5(d) which are the enlarged details of the fifth frame, it can be seen that GMM-wiener has a slight blocking artifact due to block processing.

The second experiment is in a visible wave band. We slightly adjust the setup in Fig. 4. The source is changed to a visible light source. The NIR filter is removed. Figure 6 presents the reconstructions in a visible band. Similarly, with the TCI idea, we can obtain high-speed object images using a slow imaging sensor. The variance of noise is $\sigma^2 = 10^{-4}$. The same conclusion can be drawn that GMM-wiener can obtain more sharpness than TwIST-wiener algorithms. However, we can also find that the blocking artifact of GMM-wiener becomes obvious in a visible wave band.

In conclusion, we study TCI and build an experimental setup to demonstrate TCI for NIR and visible bands in this Letter. We reconstruct sharp 500 fps videos from coded measurement frames at 50 fps. With these results, we are confident that TCI is a good idea for fast IR imaging, in which the slow imaging speed limits its application scope. The idea can also be implemented into other bands such as SWIR, MWIR, or LWIR in which the imaging speed is slow, and per-pixel price is high. To further improve the imaging speed, we will study

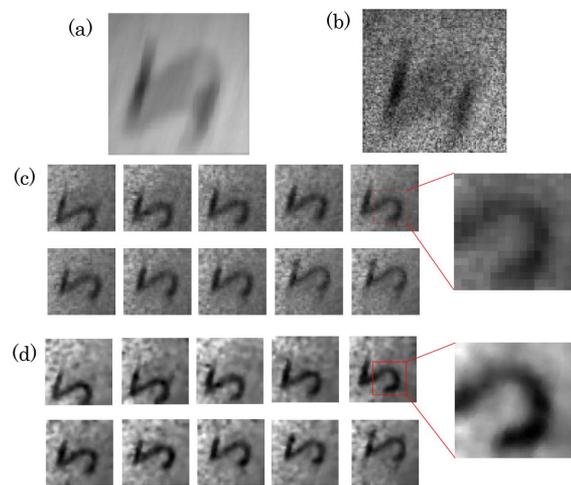


Fig. 6. Experimental results in a visible band. (a) One frame captured by a low frame rate camera; (b) one coded measurement frame; reconstructions using (c) GMM-wiener and (d) TwIST-wiener. The complete results can be found in [Visualization 14](#).

other reconstruction methods, including deep learning methods in the future.

Funding. National Natural Science Foundation of China (NSFC) (11727801, 61307022, 61675023).

REFERENCES

1. D. Reddy, A. Veeraraghavan, and R. Chellappa, *IEEE Computer Vision and Pattern Recognition (CVPR)* (2011), p. 329.
2. P. Llull, X. Liao, X. Yuan, J. Yang, D. Kittle, L. Carin, G. Sapiro, and D. J. Brady, *Opt. Express* **21**, 10526 (2013).
3. Y. Hitomi, J. Gu, M. Gupta, and S. K. Nayar, *IEEE Int. Conf. Comput. Vis.* **47**, 287 (2011).
4. J. M. Bioucas-Dias and M. A. T. Figueiredo, *IEEE Trans. Image Process.* **16**, 2992 (2007).
5. X. Liao, H. Li, and L. Carin, *SIAM J. Imag. Sci.* **7**, 797 (2014).
6. J. Yang, X. Yuan, X. Liao, P. Llull, D. J. Brady, G. Sapiro, and L. Carin, *IEEE Trans. Image Process.* **23**, 4863 (2014).
7. X. Yuan, J. Yang, P. Llull, G. Sapiro, D. J. Brady, and L. Carin, *IEEE International Conference on Image Processing (IEEE, 2013)*, p. 14.
8. X. Yuan, Y. Sun, and S. Pang, *Appl. Opt.* **56**, 2697 (2017).
9. M. Iliadis, L. Spinoulas, and A. K. Katsaggelos, *Digit. Signal Process.* **72**, 9 (2017).
10. A. Chambolle, *J. Math. Imaging Vis.* **20**, 73 (2004).
11. J. Ke and E. Y. Lam, *Opt. Express* **20**, 22102 (2012).
12. J. Ke and E. Y. Lam, *Opt. Express* **24**, 9869 (2016).