Coded aperture compressive temporal imaging

Patrick Llull, Xuejun Liao, Xin Yuan, Jianbo Yang, David Kittle, Lawrence Carin, Guillermo Sapiro, and David J. Brady*

Fitzpatrick Institute for Photonics, Department of Electrical and Computer Engineering
Duke University, 129 Hudson Hall, Durham, North Carolina 27708*, USA*

**dbrady@duke.edu*

Abstract: We use mechanical translation of a coded aperture for code division multiple access compression of video. We discuss the compressed video's temporal resolution and present experimental results for reconstructions of > 10 frames of temporal data per coded snapshot.

OCIS codes: (110.1758) Computational imaging; (100.3010) Image reconstruction techniques; (110.6915) Time imaging; Compressive sampling.

References and links

1. D. J. Brady, M. E. Gehm, R. A. Stack, D. L. Marks, D. S. Kittle, D. R. Golish, E. M. Vera, and S. D. Feller, "Multiscale gigapixel photography," Nature **486**(7403), 386–389, (2012).
2. S. Kleinfelder, S. H. Lim, X. Liu, and A. El Gamal, "A 10000 frames/s CMOS digital pixel sensor," IEEE J. Solid-St. Circ. **36**(12), 2049–2059, (2001).
3. R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan, "Light field photography with a hand-held plenoptic camera," Computer Science Technical Report CSTR 2, (2005).
4. D. J. Brady, *Optical Imaging and Spectroscopy*. (Wiley-Interscience, 2009).
5. D. J. Brady, M. Feldman, N. Pitsianis, J. P Guo, A. Portnoy, and M. Fiddy, "Compressive optical MONTAGE photography," Photonic Devices and Algorithms for Computing VII 5907(1), 590708, (2005).
6. M. Shankar, N. P. Pitsianis, and D. J. Brady, "Compressive video sensors using multichannel imagers," Appl. Opt. **49**(10), B9–B17, (2010).
7. Y. Hitomi, J. Gu, M. Gupta, T. Mitsunaga, and S. K. Nayar, "Video from a single coded exposure photograph using a learned over-complete dictionary," in *Proceedings of IEEE International Conference on Computer Vision*. (IEEE, 2011), pp. 287–294.
8. D. Takhar, J. N. Laska, M. B. Wakin, M. F. Duarte, D. Baron, S. Sarvotham, K. F. Kelly, and R. G. Baraniuk, "A new compressive imaging camera architecture using optical-domain compression," Proc. SPIE **6065**, 606509 (2006).
9. M. Wakin, J. Laska, M. Duarte, D. Baron, S. Sarvotham, D. Takhar, K.F. Kelly, and R. G. Baraniuk, "Compressive imaging for video representation and coding," in *Proceedings of Picture Coding Symposium*, (2006).
10. Y. Oike and A. E Gamal, "A 256$\times$ 256 CMOS image sensor with $\delta\sigma$-based single-shot compressed sensing," In *Proceedings of IEEE International Solid-State Circuits Conference Digest of Technical Papers* (IEEE, 2012), pp. 386–388.
11. M. Zhang and A. Bermak, "CMOS image sensor with on-chip image compression: A review and performance analysis," J. Sens. **2010**, 1–17, (2010).
12. A. Fish and O. Yadid-Pecht, "Low Power CMOS Imager Circuits," in *Circuits at the Nanoscale: Communications, Imaging, and Sensing*, K. Iniewski ed. (CRC Press, Inc., 2008), pp. 457–484.
13. V. Treeaporn, A. Ashok, and M. A. Neifeld, "Space–time compressive imaging," App. Opt. **51**(4), A67–A79, (2012).
14. M. A. Neifeld and P. Shankar, "Feature-specific imaging," App. Opt. **42**(17), 3379–3389, (2003).
15. E. J. Candès and T. Tao, "Reflections on compressed sensing," IEEE Information Theory Society Newsletter **58**(4), 20–23, (2008).
16. D. L. Donoho, "Compressed sensing," IEEE Trans. Inf. Theory **52**(4), 1289–1306, (2006).
17. R. Raskar, A. Agrawal, and J. Tumblin, "Coded exposure photography: motion deblurring using fluttered shutter," ACM Transactions on Graphics **25**(3), 795–804, (2006).

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013

18. D. Reddy, A. Veeraraghavan, and R. Chellappa, "P2C2: Programmable pixel compressive camera for high speed imaging," In Proceedings of IEEEE Conference on Coputer Vision and Pattern Recognition (IEEE, 2011), pp. 329–336.
19. A. C. Sankaranarayanan, C. Studer, and R. G. Baraniuk. "CS-MUVI: Video compressive sensing for spatial-multiplexing cameras," In *Proceedings of IEEE International Conference on Computational Photography* (IEEE, 2012), pp. 1–10.
20. M. E. Gehm, R. John, D. J. Brady, R. M. Willett, and T. J. Schulz, "Single-shot compressive spectral imaging with a dual-disperser architecture," Optics Express **15**(21), 14013–14027, (2007).
21. X. Liao, H. Li, and L. Carin, "Generalized alternating projection for weighted-$\ell_{2,1}$ minimization with applications to model-based compressive sensing," SIAM Journal on Imaging Sciences (to be published).
22. J. M. Bioucas-Dias, and M. A. T. Figueiredo, "A new TwIST: Two-step iterative shrinkage/thresholding algorithms for image restoration," IEEE Transactions on Image Processing **16**(12), 2992–3014 (2007).

1. Introduction

Cameras capturing $>$ 1 gigapixel [1], $>$ 10,000 frames per second [2], $>$ 10 ranges [3] and $>$ 100 color channels [4] demonstrate that the optical data stream entering an aperture may approach 1 exapixel/second. While simultaneous capture of the all optical information is not precluded by diffraction or quantum limits, this capacity is beyond the capability of current electronics. Previous compressive sampling proposals to reduce read-out bandwidth paradoxically increase system volume[5, 6] or operating bandwidth [7–9]. Here we use mechanical translation of a coded aperture to compress temporal sampling by $>$ 10x without substantially increasing system volume or power. The coded aperture enables spatial compression on the order of the size of the code feature relative to the detector pixel pitch. We describe computational estimation of 14 unique, high-speed temporal frames from a single captured frame. By combining physical layer compression as demonstrated here with sensor-layer compressive sampling strategies [10, 11], and by using multiscale design to parallelize read-out, one may imagine streaming full exapixel data cubes over practical communications channels.

Under ambient illumination, cameras commonly capture 10-1000 pW per pixel, corresponding to 10^9 photons per second per pixel. At this flux, frame rates approaching 10^6 per second may still provide useful information. Unfortunately, frame rate generally falls well below this limit due to read-out electronics. The power necessary to operate an electronic focal plane is proportional to the rate at which pixels are read-out [12]. Current technology requires pW to nW per pixel, implying 1-100W per sq. mm of sensor area for full data optical data cube capture. This power requirement cascades as image data flows through pipeline of read-out, processing, storage, communications, and display. Given the need to maintain low focal plane temperatures, this power density is unsustainable.

While interesting features persist on all resolvable spatial and temporal scales, the true information content of natural fields is much less than the photon flux because diverse spatial, spectral and temporal channels contain highly correlated information [13]. Under these circumstances, feature specific [14] and compressive [15, 16] multiplex measurement strategies developed over the past decade have been shown to maintain image information even when the number of digital measurement values is substantially less than the number of pixels resolved. Compressive measurement for visible imaging has been implemented using Liquid Crystal on Silicon (LCoS) devices to code pixel values incident on a single detector [8, 9]. Unfortunately, this strategy increases, rather than decreases, operating power and bandwidth because the increased data load in the encoding signal is much greater than the decreased data load of the encoded signal. If the camera estimates F frames with N pixels per frame from M measurements, then MN control signals enter the modulator to obtain FN pixels. The bandwidth into the compressive camera exceeds the output bandwidth needed by a conventional camera by the factor NC, where C is the compression ratio. Unless $C < 1/N$, implying that the camera takes less than one measurement per frame, the control bandwidth exceeds the bandwidth necessary

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013

to fully read a conventional imager. This problem is slightly less severe in coding strategies derived from "flutter shutter" [17] motion compensation strategies. The flutter shutter uses full frame temporal modulation to encode motion blur for inversion. Several studies have implemented per-pixel flutter shutter using spatial light modulators for video compression[7, 18, 19]. If we assume that these systems reconstruct at the full frame rate of the modulator, the control bandwidth is exactly equal to the bandwidth needed to read a conventional camera operating at the decompressed framerate. Alternatively, one may entirely avoid these problems by using parallel camera arrays with independent per pixel codes [5, 6] at the cost of increasing camera volume and cost by a factor of M.

Here we propose mechanical translation of a passive coded aperture for low power space-time compressive measurement. Coding is implemented by a chrome-on-glass binary transmission mask in an intermediate image plane. In contrast with previous approaches, modulation of the image data stream by harmonic oscillation of this mask requires no code transmission or operating power. We have previously used such masks for compressive imaging in coded aperture snapshot spectral imagers (CASSI) [20], which include an intermediate image plane before a spectrally dispersive relay optic. Here we demonstrate coded aperture compressive temporal imaging (CACTI). CASSI and CACTI share identical mathematical forward models. In CASSI, each plane in the spectral datacube is modulated by a shifted code. Dispersion through a grating or prism shifts spectral planes after coded aperture modulation. Detection integrates the spectral planes, but the datacube can be recovered by isolating each spectral plane based on its local code structure. This process may be viewed as code division multiple access (CDMA). In CACTI, translation of the coded aperture during exposure means that each temporal plane in the video stream is modulated by a shifted version of the code, thereby attaining per-pixel modulation using no additional sensor bandwidth.

Signal separation once again works by CDMA. We isolate the object's temporal channels from the compressed data by inverting a highly-underdetermined system of equations. By using an iterative reconstruction algorithm, we may estimate several high-speed video frames from a single coded measurement.

2. Theory

One may view CACTI's CDMA sensing process as uniquely patterning high-speed spatiotemporal object voxels $f(x,y,t) \in \mathbb{R}^3$ with a transmission function that shifts in time (Fig. 1). Doing this applies distinct local coding structures to each temporal channel prior to integrating the channels as limited-framerate images $g(x',y',t') \in \mathbb{R}^2$ on the N-pixel detector. An N_F-frame, high-speed estimate of $f(x,y,t)$ may be reconstructed from each low-speed coded snapshot $g(x',y',t')$, with $t' < t$.

Considering only one spatial dimension ($(x,y) \rightarrow x$) and respectively denoting object-and image-space coordinates with unprimed and primed variables, the sampled data $g(x',t')$ consists of discrete samples of the continuous transformation [4]

$$g(x',t') = \int_1^{N_F} \int_1^{N} f(x,t) T(x - s(t)) \text{rect}\left(\frac{x-x'}{\Delta_x}\right) \text{rect}\left(\frac{t-t'}{\Delta_t}\right) dx dt, \tag{1}$$

where $T(x-s(t))$ represents the transmission function of the coded aperture, Δ_x is the detector pixel size, $\text{rect}(\frac{x}{\Delta_x})$ is the pixel sampling function and Δ_t is the temporal integration time. $s(t)$ describes the coded aperture's spatial position during the camera's integration window.

One may analyze the expected temporal resolution of the coded data by considering the Fourier transform of Eq. (1). Assuming the coded aperture moves linearly during Δ_t such that $s(t) = vt$, the image's temporal spectrum is given by

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013

$$\hat{g}(u,v) = \mathrm{sinc}(u\Delta_x)\mathrm{sinc}(v\Delta_t) \int \hat{f}(u-w, v-\nu w)\hat{T}(w)dw, \tag{2}$$

where $\hat{f}(u,v)$ is the 2D Fourier transform of the space-time datacube and $\hat{T}(w)$ is the 1D Fourier transform of the spatial code. Without the use of the coded aperture, $\hat{g}(u,v) = \mathrm{sinc}(u\Delta_x)\mathrm{sinc}(v\Delta_t)\hat{f}(u,v)$ and the sampled data stream is proportional to the object video low-pass filtered by the pixel sampling functions. Achievable resolution is proportional to Δ_x in x and Δ_t in time. The moving code aliases higher frequency components of the object video into the passband of the detector sampling functions. The support of $\hat{T}(w)$ extends to some multiple of the code feature size Δ_c (in units of detector pixels), meaning that the effective passband may be increased by a factor proportional to $1/\Delta_c$ in space and ν/Δ_c in time. In practice, finite mechanical deceleration times cause $\hat{T}(w)$ to have significant DC and low-frequency components in addition to the dominant $\nu = \frac{C}{\Delta_t}$; hence, high and low frequencies alike are aliased into the system's passband.

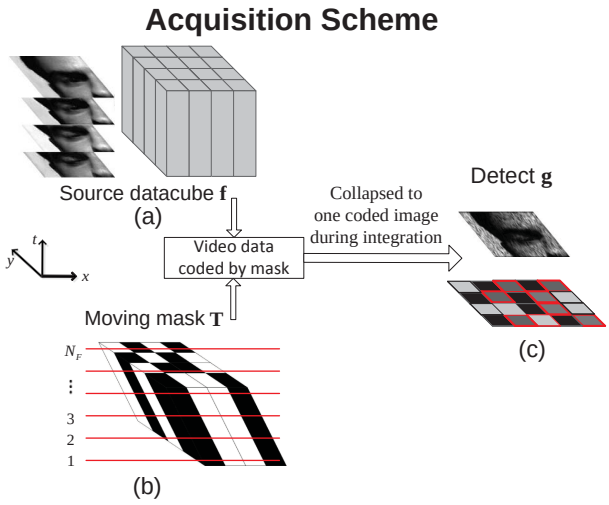


Fig. 1. CACTI image acquisition process. (a) A discrete space-time source datacube is (b) multiplied at each of N_F temporal channels with a shifted version of a coded aperture pattern. (c) Each detected frame $\mathbf{g}$ is the summation of the coded temporal channels and contains the object's spatiotemporal-multiplexed information. The dark grey (red-outlined) and black detected pixels in (c) pictorially depict the code's location at the beginning and the end of the camera's integration window, respectively.

Considering a square N-pixel active sensing area, the discretized form of the three-dimensional scene is $\mathbf{f} \in \mathbb{R}^{\sqrt{N}\times\sqrt{N}\times N_F}$, i.e. a $(\sqrt{N}\times\sqrt{N})\times N_F$-voxel spatiotemporal datacube. In CACTI, a time-varying spatial transmission pattern $\mathbf{T} \in \mathbb{R}^{\sqrt{N}\times\sqrt{N}\times N_F}$ uniquely codes each of the N_F temporal channels of $\mathbf{f}$ prior to integrating them into one detector image $\mathbf{g} \in \mathbb{R}^{\sqrt{N}\times\sqrt{N}}$ during Δ_t. These measurements at spatial indices (i,j) and temporal index k are given by

$$g_{i,j} = \sum_{k=1}^{N_F} T_{i,j,k} f_{i,j,k} + n_{i,j}, \tag{3}$$

where $n_{i,j}$ represents imaging noise at the $(i,j)^{th}$ pixel. One may rasterize the discrete object

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013

$\mathbf{f} \in \mathbb{R}^{NN_F \times 1}$, image $\mathbf{g} \in \mathbb{R}^{N \times 1}$, and noise $\mathbf{n} \in \mathbb{R}^{N \times 1}$ to obtain the linear transformation given by

$$\mathbf{g} \quad = \quad \mathbf{Hf} + \mathbf{n}, \tag{4}$$

where $\mathbf{H} \in \mathbb{R}^{N \times NN_F}$ is the system's *discrete forward matrix* that accounts for sampling factors including the optical impulse response, pixel sampling function, and time-varying transmission function. The forward matrix is a 2-dimensional representation of the 3-dimensional transmission function $\mathbf{T}$:

$$\mathbf{H}_k \quad \stackrel{\text{def}}{=} \quad \text{diag}\left[T_{1,1,k} \;\; T_{2,1,k} \;\; \cdots \;\; T_{\sqrt{N},\sqrt{N},k}\right], \quad k = 1, \ldots, N_F; \tag{5}$$

$$\mathbf{H} \quad \stackrel{\text{def}}{=} \quad [\mathbf{H}_1 \;\; \mathbf{H}_2 \;\; \cdots \;\; \mathbf{H}_{N_F}], \tag{6}$$

where $\mathbf{H}_k \in \mathbb{R}^{N \times N}$ is a matrix containing the entries of $\mathbf{T}_k$ along its diagonal and $\mathbf{H}$ is a concatenation of all $\mathbf{H}_k$, $k \in \{1, \ldots, N_F\}$. Figure 2 underlines the role $\mathbf{H}$ plays in the linear transformation.

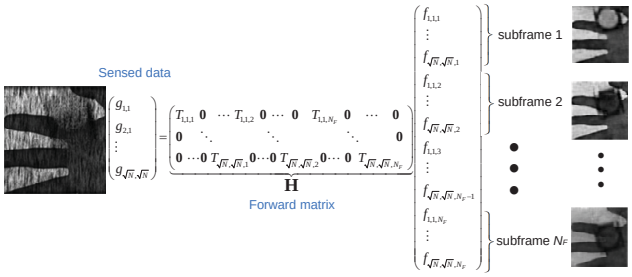


Fig. 2. Linear system model. N_F subframes of high-speed data $\mathbf{f}$ are estimated from a single snapshot $\mathbf{g}$. The forward model matrix $\mathbf{H}$ has many more columns than rows and has dimensions $N \times (N \times N_F)$.

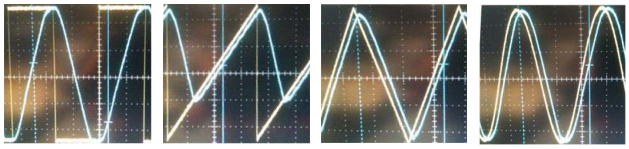


Fig. 3. Waveform choices for $s(t)$. Yellow: signal from function generator. Blue: actual hardware motion. Note the poor mechanical response to sharp rising/falling edges in (a) and (b). The sine wave (d) is unpreferable because of the nonuniform exposure time of different $\mathbf{T}_k$.

At the k^{th} temporal channel, the coded aperture's transmission function $\mathbf{T}$ is given by

$$\mathbf{T}_k = \text{Rand}(\sqrt{N}, \sqrt{N}, s_k), \tag{7}$$

where $\text{Rand}(m, n, p)$ denotes a 50%, $m \times n$ random binary matrix shifted vertically by p pixels (optimal designs could be considered for this system as well). s_k discretely approximates $s(t)$ at the k^{th} temporal channel by

$$s_k = C\text{Tri}\left[\frac{k}{2\Delta_t}\right] \tag{8}$$

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013

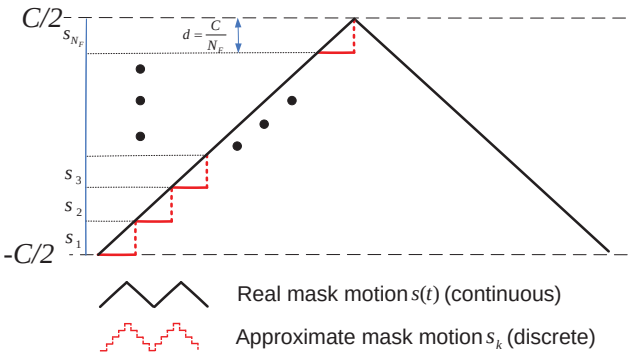


Fig. 4. Continuous motion and discrete approximation to coded aperture movement during integration time. The discrete triangle function s_k more-accurately approximates the continuous triangle wave driving the mask with smaller values of d but adds more columns to **H**.

where C, the system's compression ratio, is the amplitude traversed by the code in units of detector pixels. Tri $\left[\frac{k}{2\Delta_t}\right]$ represents a discrete triangle wave signal of twice the integration time periodicity. The camera integrates during the C-pixel sweep of the coded aperture on the image plane and detects a linear combination of C uniquely-coded temporal channels of f.

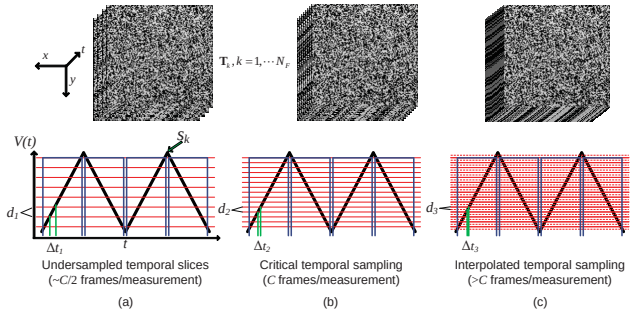


Fig. 5. Temporal channels used for the reconstruction. Red lines indicate which subset of transverse mask positions s_k were utilized to construct the forward matrix **H**. Blue lines represent the camera integration windows. (a) Calibrating with fewer s_k results in a better-posed inverse problem but doesn't as closely approximate the temporal motion $s(t)$. (b) With $d = 1$, each pixel integrates several unique coding patterns with a temporal separation $\Delta t = C^{-1}$. (c) Constructing **H** with large N_F ($d < 1$) interpolates the motion occurring between the uniquely-coded image frames but retains most of the residual motion blur.

Periodic triangular motion lets us use the same discrete forward matrix **H** to reconstruct any given snapshot **g** within the acquired video while adhering to the hardware's mechanical acceleration limitations (Fig. 3). The discrete motion s_k (Eq. (8)) closely approximates the analog triangle waveform supplied by the function generator (Fig. 4).

Let d represent the number of *detector* pixels the mask moves between adjacent temporal channels s_k and s_{k+1}. N_F frames are reconstructed from a single coded snapshot given by

$$N_F = \frac{C}{d}, \tag{9}$$

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013

thus, altering d will affect the number of reconstructed frames for given a compression ratio C.

In the case of $d = 1$ (i.e. $N_F = C$), the detector pixels that sense the continuous, temporally-modulated object f are *critically-encoded*; each pixel integrates a series of nondegenerate mask patterns (Fig. 5(b)) during Δ_t.

When $d < 1$ (i.e. $N_F > C$), every $\frac{1}{d}$ temporal channels of $\mathbf{H}$ will contain nondegenerate temporal code information. These channels will reconstruct as if the sensing pixels are critically-encoded. The other temporal slices will interpolate the motion *between* critically-encoded temporal channels (Fig. 5(c)). Generally, this interpolation accurately estimates the direction of the motion between these critically-encoded frames but retains most of the residual motion blur. Although it is difficult to see how this form of interpolation affects the temporal resolution of g, one may use it to smoothen the reconstructed frames as evident in the experimental videos as presented in Section 5.

3. Experimental hardware

The experimental prototype camera (Fig. 6) consists of a 50mm camera objective lens (Computar), a lithographically-patterned chrome-on-quartz coded aperture with anti-reflective coating for visible wavelengths [20] (Eq. (7)) mounted upon a piezoelectric stage (Newport Co.), an $F/8$ achromatic relay lens (Edmund Optics), and a 640×480 FireWire IEEE 1394a monochrome CCD camera (Marlin AVT).

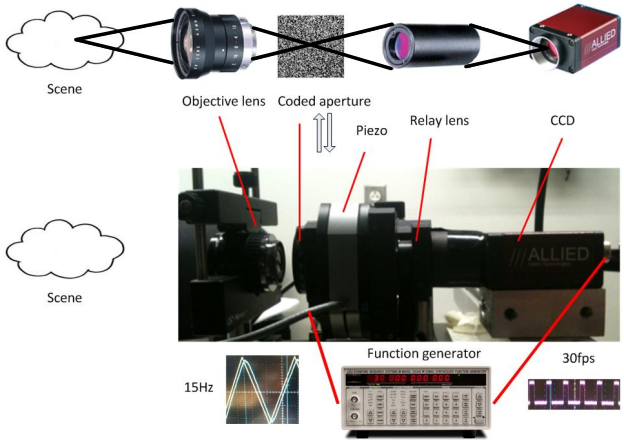


Fig. 6. CACTI Prototype hardware setup. The coded aperture is 5.06mm $\times$ 4.91mm and spans 248 $\times$ 256 detector pixels. The function generator moves the coded aperture and triggers camera acquisition with signals from its function and SYNC outputs, respectively.

The objective lens images the continuous scene f onto the piezo-positioned mask. The function generator (Stanford Research Systems DS345) drives the piezo with a 10V pk-pk, 15Hz triangle wave to locally code the image plane while the camera integrates. We operate at this low frequency to accommodate the piezo's finite mechanical deceleration time.

To ensure the CDMA process remains time-invariant, we use the function generator's SYNC output to generate a 15Hz square wave. We frequency-double this signal using an FPGA device (Altera) to trigger camera integrations once along the mask's upward and downward motion (Fig. 5).

The relay lens images the spatiotemporally modulated scene onto the camera, which saves the 30fps coded snapshots to a local computer. N_F video frames of the discrete scene $\mathbf{f}$ are

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013

later reconstructed from each coded image $\mathbf{g}$ offline by the Generalized Alternating Projection (GAP) [21] algorithm.

During Δ_t, the piezo can move a range of $0-160\mu m$ vertically in the (x,y) plane. Using $158.4\mu m$ of this stroke moves the coded aperture eight $19.8\mu m$ elements (sixteen $9.9\mu m$ detector pixels) during each camera integration period Δ_t. Using larger strokes for a given modulation frequency is possible and would increase C.

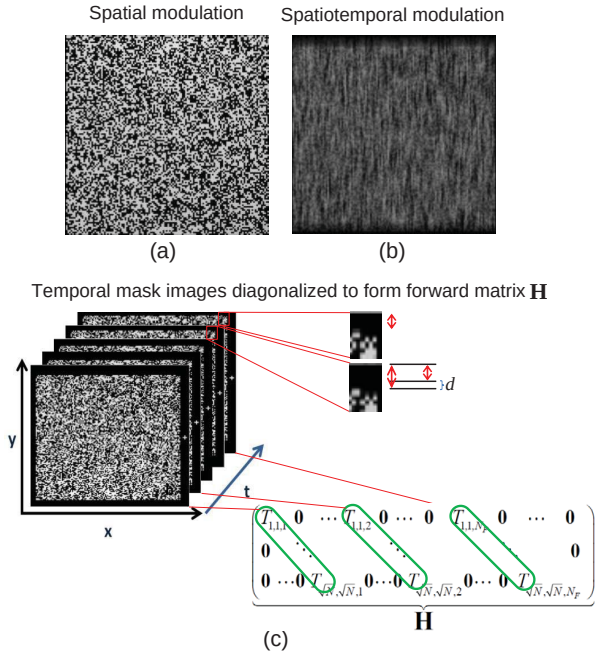


Fig. 7. Spatial and temporal modulation. (a) A stationary coded aperture spatially modulates the image. (b) Moving the coded aperture during the integration window applies local code structures to N_F temporal channels, effectively shearing the coded space-time datacube and providing per-pixel flutter shutter. (c) Imaging the (stationary) mask at positions d pixels apart and storing them into the forward matrix $\mathbf{H}$ simulates the mask's motion, thereby conditioning the inversion.

The benefits of such a power-efficient system are numerous. Considering a standard CCD sensor may draw on the order of $1-10$W per sq. mm, compressing the optical data stream's temporal channels grants the system designer greater liberty to use larger detectors, smaller pixels, and assume more forgiving thermal conditions when operating at a given framerate.

The piezoelectric stage draws an average of .2W of power to modulate approximately $N = 65,000$ pixels. Importantly, when operating at a given framerate, CACTI's passive coding scheme facilitates scalability without increasing power usage; one may simply use a larger coded aperture to modulate larger values of N pixels with negligible additional on-board power overhead. This passive coding scheme holds other advantages, such as compactness and polarization independence, over reflective LCoS-based modulation strategies, whereby the additional bandwidth required to modulate the datacube increases proportionally to N.

Using this piezoelectric stage itself is not the optimal solution to translate a coded aperture during Δ_t. This device was preferable for the hardware prototype because of its precision and convenient built-in Matlab interface. However, a low-resistance spring system could, in princi-

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013

ple, serve the same purpose while using very little power.

3.1. Forward model calibration

We calibrate the system forward model by imaging the aperture code under uniform, white illumination at discrete spatial steps s_k according to Eq. (8). Steps are d detector pixels apart over the coded aperture's range of motion (Fig. 7(c), Fig. 5,). This accounts for system misalignments and relay-side aberrations. A Matlab routine controls the piezoelectric stage position during calibration. Since Matlab cannot generate a near-analog waveform for continuous motion, we connect the piezoelectric motion controller to the function generator via serial port during experimental capture.

We use an active area of 281×281 detector pixels to account for the 248×256-pixel coded aperture's motion s_k with additional zero-padding. We choose $d = \frac{\Delta_x}{10}$ to provide a substantial basis with which to construct the forward model while remaining well above the piezo's 0.0048 pixel RMS jitter. Storing every temporal channel spaced $d = 0.99\mu m$ apart into $\mathbf{H}$ results in $N_F = 160$ reconstructed frames.

When reconstructing, we may diagonalize any subset of temporal slices of the $281 \times 281 \times 160$ set of mask images into the forward model (Fig. 7). We found the optimum subset of mask positions within this 160-frame set of s_k through iterative $||\mathbf{g} - \mathbf{H}\mathbf{f_e}||_2$-error reconstruction tests, where $\mathbf{f_e}$ is GAP's N_F-frame estimate of the continuous motion f. From these tests, we chose and compared two numbers of frames to reconstruct per measurement, $N_F = C = 14$ and $N_F = 148$. $\mathbf{H}$ has dimensions $281^2 \times (281^2 \times N_F)$ for both of these cases.

As seen in Figs. 10-13, decreasing d and estimating up to 148 frames from a single exposure $\mathbf{g}$ does not significantly reduce the aesthetic quality of the inversion results, nor does it significantly affect the residual error (Fig. 8(b)). The reconstruction time increases approximately linearly with N_F as shown in Fig. 8(a).

4. Reconstruction algorithms

Since $\mathbf{H}$ multiplexes many local code patterns of the continuous object to the discrete-time image $\mathbf{g}$, inverting Eq. (4) for $\mathbf{f}$ becomes difficult as N_F increases. Least-squares, pseudoinverse, and other linear inversion methods cannot accurately reconstruct such underdetermined systems.

Modern reconstruction methods, such as Bayesian, GMM, dictionary-based, and maximum likelihood algorithms are capable of decompressing the data with varying degrees of success based on the spatiotemporal structure of the video. Of these, we use two iterative reconstruction algorithms called Two-step Iterative-Shrinkage Thresholding (TwIST)[22] and Generalized Alternating Projection (GAP) [21], both exploiting image and video models (priors) to effectively solve this ill-posed inverse problem. TwIST applies a regularization function to penalize estimates of $\mathbf{f}$ that are unlikely or undesirable to occur in the estimated $\mathbf{f_e}$ while GAP takes advantage of the structural sparsity of the subframes in transform domains such as wavelets and discrete cosine transform (DCT).

4.1. TwIST

TwIST solves the unconstrained optimization problem

$$\mathbf{f_e} = \arg\min_{\mathbf{f}} \|\mathbf{g} - \mathbf{H}\mathbf{f}\|^2 + \lambda\Omega(\mathbf{f}), \tag{10}$$

where $\Omega(\mathbf{f})$ and λ are the regularizer and regularization weights, respectively [22]. The regularizer penalizes characteristics of the estimated $\mathbf{f}$ that would result in poor reconstructions. The

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013

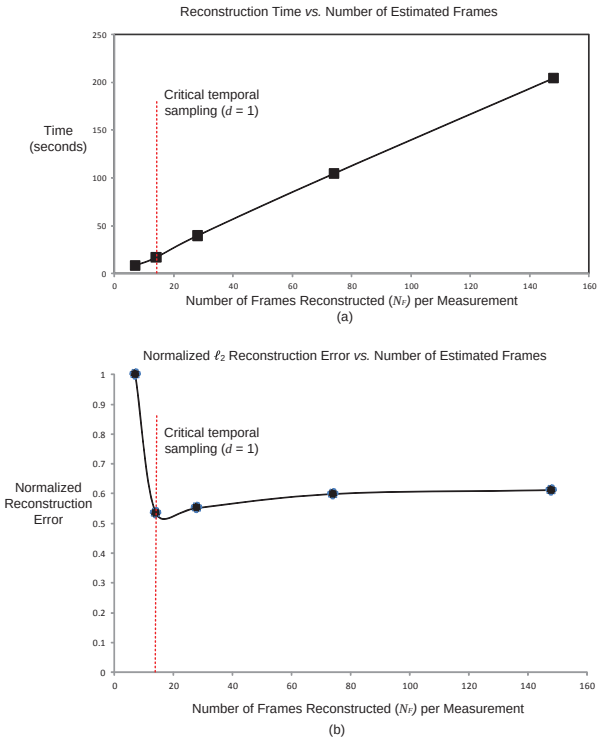


Fig. 8. Algorithm convergence times and relative residual reconstruction errors for various compression ratios. (a) GAP's reconstruction time increases linearly with data size. Tests were performed on a ASUS U46E laptop (Intel quad core I7 operated at 3.1GHz. (b) Normalized ℓ_2 reconstruction error vs. number of reconstructed frames. The residual error reaches a minimum at critical temporal sampling and gradually flattens out with finer temporal interpolation (lower d).

reconstruction results presented in Section 5 employ a Total Variation (TV) regularizer given by

$$\Omega(\mathbf{f}) = \sum_{k}^{N_F} \sum_{i,j}^{N} \sqrt{(f_{i+1,j,k} - f_{i,j,k})^2 + (f_{i,j+1,k} - f_{i,j,k})^2}, \tag{11}$$

and hence penalize estimates with sharp spatial gradients. Because of this, sparsity in spatial gradients within each temporal channel is enforced through the iterative process of estimating $\mathbf{f}$. We choose TV regularization since many natural scenes are well-described by sparse gradients. The regularization weight was chosen via experimental optimization over several test values $\lambda \in [0.3, 2]$. A weight of $\lambda = 1$ yielded in the clearest reconstructions and was used for the estimates obtained by TwIST presented in Section 5.

4.2. Generalized alternating projection (GAP)

In this section, we describe GAP with terminology adapted to CACTI's sensing paradigm. We exploit GAP's fast convergence time in most of the results presented in this paper and hence provide a more detailed explanation of its methods. Figure 9 illustrates the underlying principle

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013

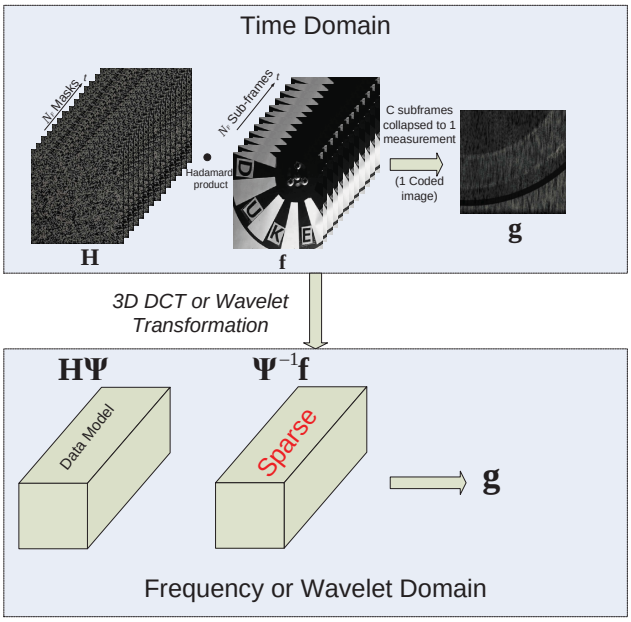


Fig. 9. Illustration of the GAP algorithm.

of GAP. It is worth noting that GAP is based on the volume's global sparsity and requires no training whatsoever. In other words, GAP is a universal reconstruction algorithm insensitive to data being inverted.

The GAP algorithm makes use of Euclidean projections on two convex sets, which respectively enforce data fidelity and structural sparsity. Furthermore, GAP is an anytime algorithm; the results produced by the algorithm converge monotonically to the true value as the computation proceeds. The monotonicity has been generally observed in our extensive experiments and theoretically established under a set of sufficient conditions on the forward model. The reconstructed subframes continually improve over successive iterations and the user can halt computation at anytime to obtain intermediate results. The user may then continue improving reconstruction by resuming the computation. The following section reviews the main steps of the GAP algorithm [21].

4.2.1. The linear manifold

Data fidelity is ensured by projection onto $\Pi = \left\{\mathbf{f} : \sum_{k=1}^{N} T_{i,j,k} f_{i,j,k} = g_{i,j},\ \forall\,(i,j)\right\}$, the linear manifold consisting of all legitimate high-speed frames that can integrate into the measured snapshot $\mathbf{g}$ by following the forward model in Eq. (4). In other words, Π is the set of solutions to an underdetermined system of linear equations, which are to be disambiguated by exploiting structural sparsity of $\mathbf{f}$ in transform domains.

4.2.2. The weighted $\ell_{2,1}$ ball

Let Ψ_1,Ψ_2,Ψ_3 be the matrices of orthonormal transforms along the two spatial coordinates and the temporal coordinate, respectively. The frames $\mathbf{f}$ are represented in $\Psi = (\Psi_1, \Psi_2, \Psi_3)$ as

$$w_{i,j,k} = [\Psi(\mathbf{f})]_{i,j,k} = \sum_{i',j',k'} \Psi_1(i,i')\Psi_2(j,j')\Psi_3(k,k') f_{i',j',k'}.$$

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013

We group the transform coefficients $\{w_{i,j,k}\}$ into m disjoint subsets, $\{w_{i,j,k} : (i,j,k) \in G_l\}$, $l = 1, \cdots, m$, and weight each group G_l by a positive number β_l, where $G = \{G_1, G_2, \cdots, G_m\}$ is a partition of the coefficient indices. The weights $\{\beta_l\}_{l=1}^m$ are chosen to emphasize low-frequency coefficients (de-emphasize high-frequency coefficients) when Ψ is a discrete cosine transform, or emphasize coarse-scale coefficients (de-emphasize finer-scale coefficients) when Ψ is a wavelet transform.

A weighted $\ell_{2,1}$ ball of size C is defined as $\Lambda(C) = \{\mathbf{f} : \|\Psi(\mathbf{f})\|_{G\beta} \leq C\}$, where

$$\|\mathbf{w}\|_{G\beta} \quad = \quad \sum_{l=1}^{m} \beta_l \|\mathbf{w}_{G_l}\|_2,$$

$\|\cdot\|_2$ is standard ℓ_2 norm, and $\mathbf{w}_{G_l} = [w_{i,j,k}]_{(i,j,k)\in G_l}$ is a subvector of $\mathbf{w}$ whose elements are indicated by indices in G_l. Note that $\Lambda(C)$ is constructed as a weighted $\ell_{2,1}$ ball in the space of transform coefficients $\mathbf{w} = \Psi(\mathbf{f})$, since structural sparsity is desired for the coefficients instead of the voxels. The ball is rotated in voxel space, due to orthonormal transform $\Psi(\cdot)$.

4.2.3. Euclidean projections

The Euclidean projection of any $\tilde{\mathbf{f}} \notin \Pi$ onto Π is given component-wise by

$$[P_\Pi(\tilde{\mathbf{f}})]_{i,j,k} \quad = \quad \tilde{f}_{i,j,k} + \frac{T_{i,j,k}}{\sum_{k'=1}^{N_f} T_{i,j,k'}^2} \left(g_{i,j} - \sum_{k'=1}^{N_f} T_{i,j,k'} \tilde{f}_{i,j,k'} \right). \tag{12}$$

The Euclidean projection of any $\mathbf{f} \notin \Lambda(C)$ onto $\Lambda(C)$ is given by

$$P_{\Lambda(C)}(\mathbf{f}) = \Psi^{-1} \left(\arg \min_{\boldsymbol{\theta}:\|\boldsymbol{\theta}\|_{G\beta} \leq C} \|\boldsymbol{\theta} - \Psi(\mathbf{f})\|_2 \right),$$

where $\|\cdot\|_2$ is standard Euclidean norm. We are only interested in $P_{\Lambda(C)}(\mathbf{f})$ when C takes the special values as considered below.

4.2.4. Alternating projection between Π and $\Lambda(C)$ with a systematically changing C

The GAP algorithm is a sequence of Euclidean projections between a linear manifold and a weighted $\ell_{2,1}$ ball that undergoes a systematic change in size. Let the projections on Π be denoted by $\{\mathbf{f}^{(t)}\}$ and the projection on $\Lambda(C^{(t)})$ be denoted by $\tilde{\mathbf{f}}^{(t)}$. The GAP algorithm starts with $\tilde{\mathbf{f}}^{(0)} = \mathbf{0}$ (corresponding $C^{(0)} = 0$), iterates between the following two steps, until $\|\mathbf{f}^{(t)} - \tilde{\mathbf{f}}^{(t)}\|_2$ converges in t.

Projection on the linear manifold.

$$\mathbf{f}^{(t)} \quad = \quad P_\Pi(\tilde{\mathbf{f}}^{(t-1)}), \quad t \geq 1,$$

with the solution given in Eq. (12).

Projection on the weighted $\ell_{2,1}$ ball of changing size.

$$\tilde{\mathbf{f}}^{(t)} \; = \; P_{\Lambda(C^{(t)})}(\mathbf{f}^{(t)}) \; = \; \Psi^{-1}\left(\boldsymbol{\theta}^{(t)}\right), \quad t \geq 1,$$

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013

where $\boldsymbol{\theta}^{(t)}$, denoting $\mathbf{w}^{(t)} = \Psi(\mathbf{f}^{(t)})$, is given component-wise by

$$\theta_{i,j,k}^{(t)} \quad = \quad w_{i,j,k}^{(t)} \max\left\{1 - \frac{\beta_l \left\|\mathbf{w}_{G_{l_{m^*+1}}}^{(t)}\right\|_2}{\beta_{l_{m^\star+1}} \left\|\mathbf{w}_{G_l}^{(t)}\right\|_2}, 0\right\}, \quad \forall\,(i,j,k) \in G_l, \quad l = 1,\cdots,m,$$

and $(l_1, l_2, \cdots, l_m)$ is a permutation of $(1, 2, \cdots, m)$ such that

$$\frac{\|\mathbf{w}_{G_{l_q}}^{(t)}\|_2}{\beta_{l_q}} \geq \frac{\|\mathbf{w}_{G_{l_{q+1}}}^{(t)}\|_2}{\beta_{l_{q+1}}}$$

holds for any $q \leq m-1$, and $m^\star = \min\{z : \text{cardinality}(\cup_{q=1}^{z} G_{l_q}) \geq \text{dimensionality}(\mathbf{g})\}$. It is not difficult to verify that the ball size $C^{(t)}$ used to derive the solution of $\boldsymbol{\theta}^{(t)}$ is

$$C^{(t)} \quad = \quad \sum_{q=1}^{m^\star} \beta_{l_q}^2 \left(\frac{\left\|\mathbf{w}_{G_{l_q}}^{(t)}\right\|_2}{\beta_{l_q}} - \frac{\left\|\mathbf{w}_{G_{l_{m^*+1}}}^{(t)}\right\|_2}{\beta_{l_{m^\star+1}}} \right),$$

which depends on the most recent projection on Π.

5. Results

CACTI's experimental temporal superresolution results are shown in Figs. 10, 11, 12, and 13 (see Media 1–8 for the complete videos). An eye blinking, a lens falling in front of a hand, a chopper wheel with the letters 'DUKE' placed on the blades, and a bottle pouring water into a cup are captured at 30fps and reconstructed with $N_F = C = 14$ and $N_F = 148$. There is little aesthetic difference between these reconstructions. In some cases, as with the lens and hand reconstruction, $N_F = 148$ appears to yield additional temporal information over $N_F = 14$. The upper-left images depict the sum of the reconstructed frames, showing the expected time-integrated snapshots acquired with a 30fps video camera lacking spatiotemporal image plane modulation.

Note that several of these features, particularly the water pouring, are hardly visible among the moving code pattern. TwIST was used to reconstruct the data shown in Figs. 12,13 since TwIST reconstructed these datasets with greater visual quality. Dynamic scenes were reconstructed with GAP using a spatial DCT basis; stationary scenes were reconstructed using a spatial wavelet basis. A temporal DCT basis was used for all results with GAP. Please click on the media object files to view the complete reconstructed videos.

The compression ratio C is 14 rather than 16 because the triangle wave's peak and trough ($\mathbf{T}s_1$ and $\mathbf{T}s_{16}$) are not accurately characterized by linear motion due to the mechanical deceleration time and were hence not placed into $\mathbf{H}$ to reduce model error.

The CACTI system captures more unique coding projections of scene information when the mask moves C pixels during the exposure (Figs. 14(b),and 14(d)) than if mask is held stationary (Figs. 14(a),and 14(c)), thereby improving the reconstruction quality for detailed scenes. The stationary binary coded aperture may completely block small features, rendering the reconstruction difficult and artifact-ridden.

Completely changing the coding patterns C times during Δ_t in hardware is only possible with adequate fill-factor employing a reflective LCoS device to address each pixel C times during the integration period. To compare the reconstruction fidelity of the low-bandwidth CACTI transmission function (Eq. (7)) with this modulation strategy, we present simulated PSNR values

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013

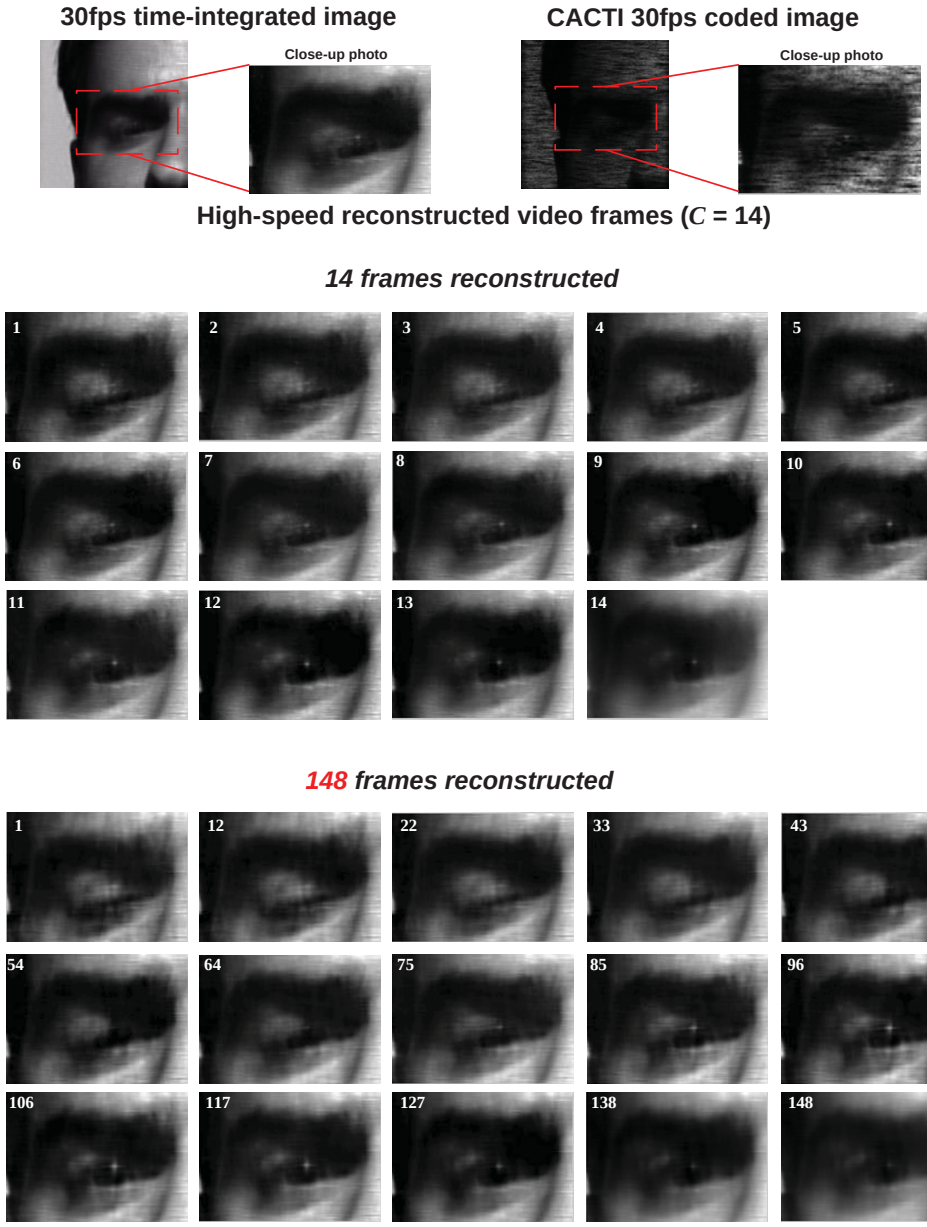


Fig. 10. High-speed ($C = 14$) video of an eye blink, from closed to open, reconstructed from a single coded snapshot for $N_F = 14$ (Media 1) and $N_F = 148$ (Media 2). The numbers on the bottom-right of the pictures represent the frame number of the video sequence. Note that the eye is the only part of the scene that moves. The top left frame shows the sum of these reconstructed frames, which approximates the motion captured by a 30fps camera without a coded aperture modulating the focal plane.

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013

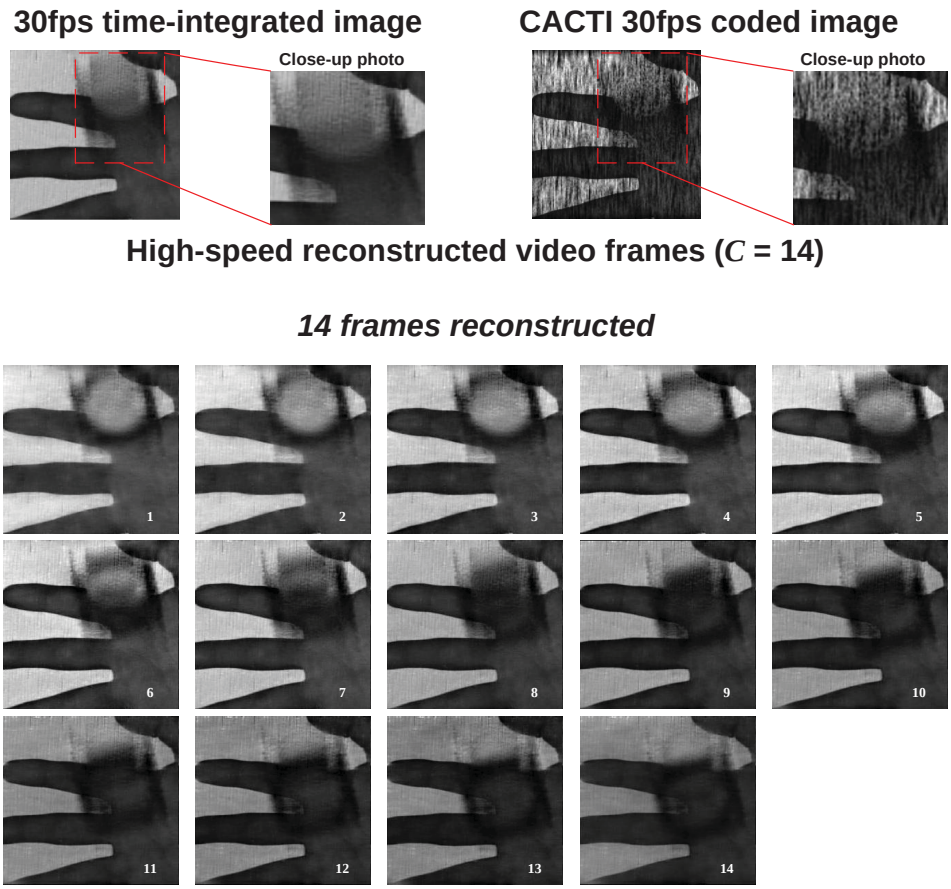
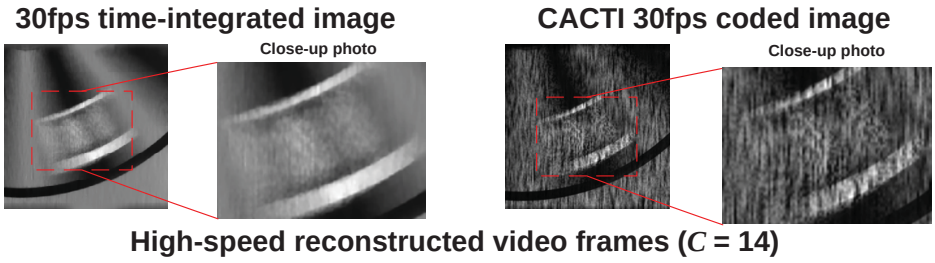


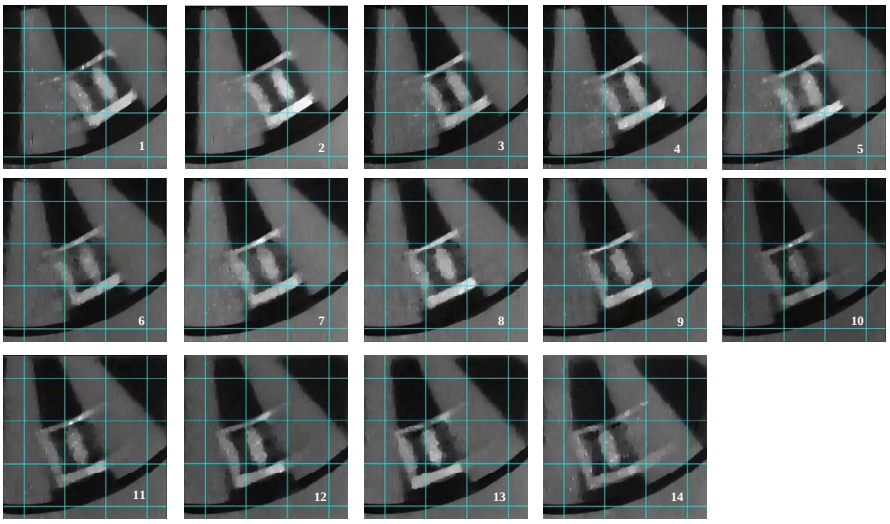
Fig. 11. Capture and reconstruction of a lens falling in front of a hand for $N_F = 14$ (Media 3) and $N_F = 148$ (Media 4). Notice the reconstructed frames capture the magnification effects of the lens as it passes in front of the hand.

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013



High-speed reconstructed video frames (C = 14)

14 frames reconstructed (grid added to help visualize movement)



148 frames reconstructed (grid added to help visualize movement)

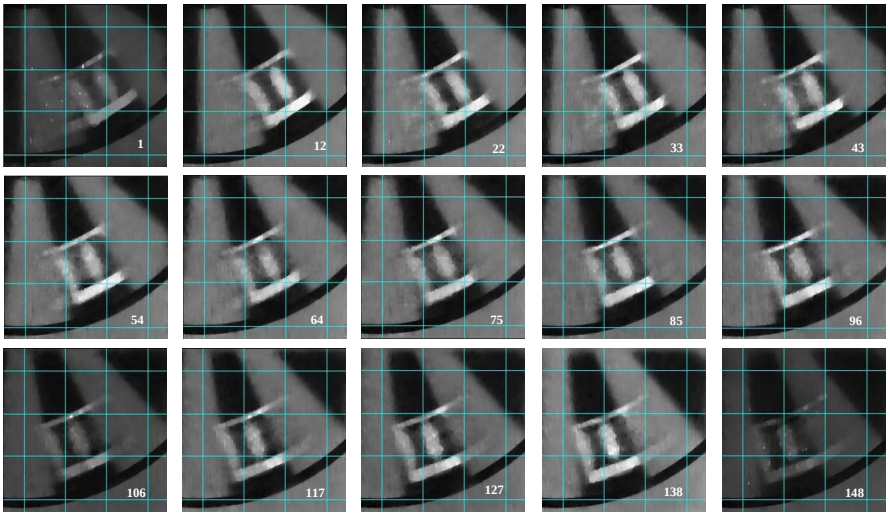


Fig. 12. Capture and reconstruction of a letter 'D' placed at the edge of a chopper wheel rotating at 15Hz for $N_F = 14$ (Media 5) and $N_F = 148$ (Media 6). The white part of the letter exhibits ghosting effects in the reconstructions due to ambiguities in the solution. The TwIST algorithm with TV regularization was used to reconstruct this data [22].

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013

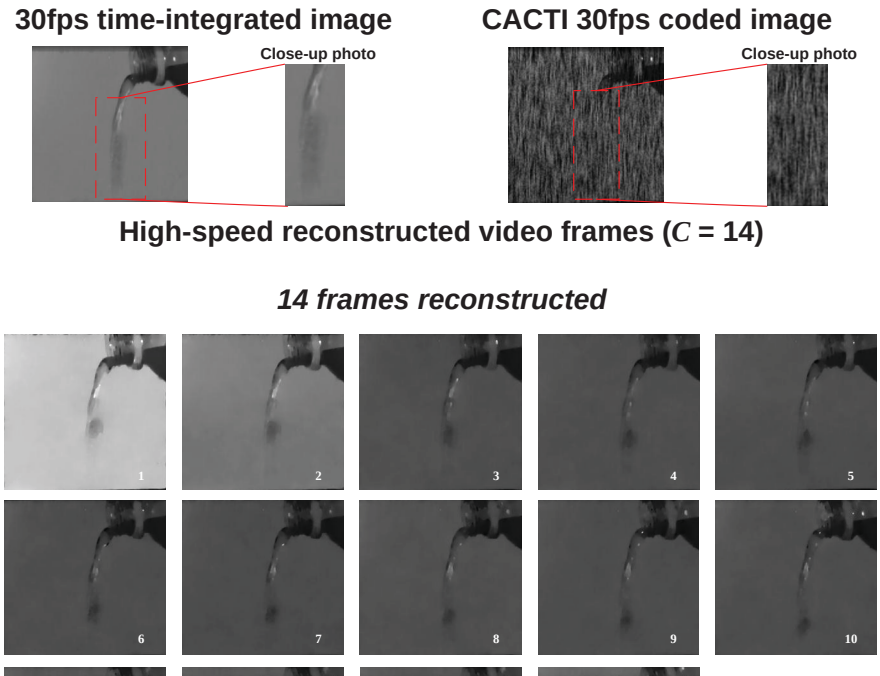


Fig. 13. Capture and reconstructed video of a bottle pouring water for $N_F = 14$ (Media 7) and $N_F = 148$ (Media 8). Note the time-varying specularities in the video. The TwIST algorithm with TV regularization was used to reconstruct this data [22].

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013

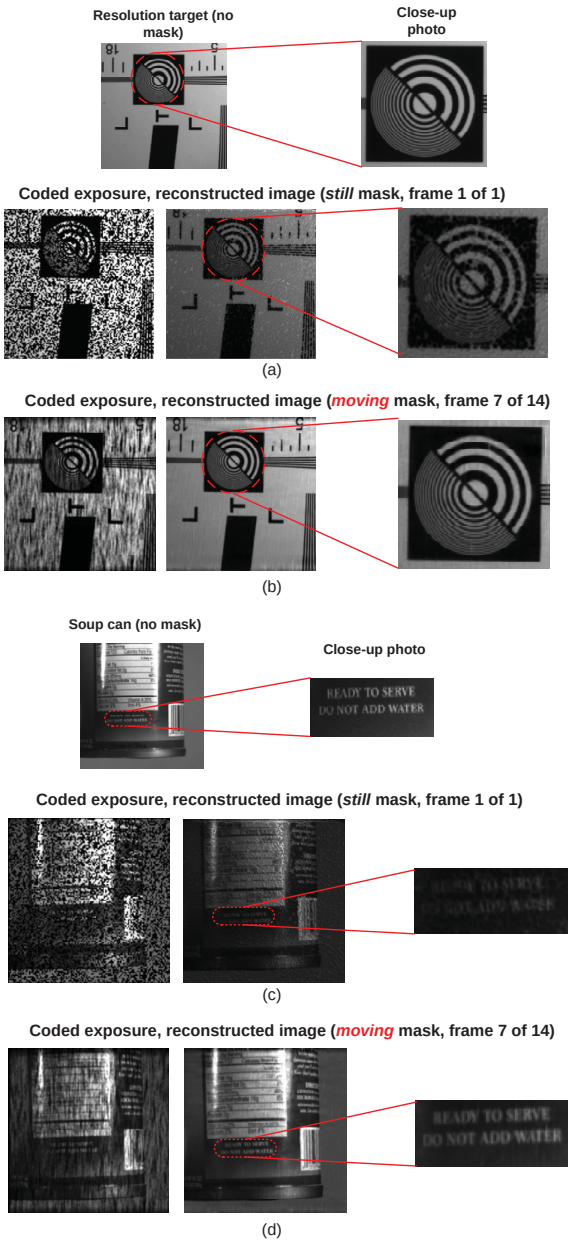


Fig. 14. Spatial resolution tests of (a),(b) an ISO 12233 resolution target and (c),(d) a soup can. These objects were kept stationary several feet away from the camera. (a),(c) show reconstructed results without temporally moving the mask; (b),(d) show the same objects when reconstructed with temporal mask motion.

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013

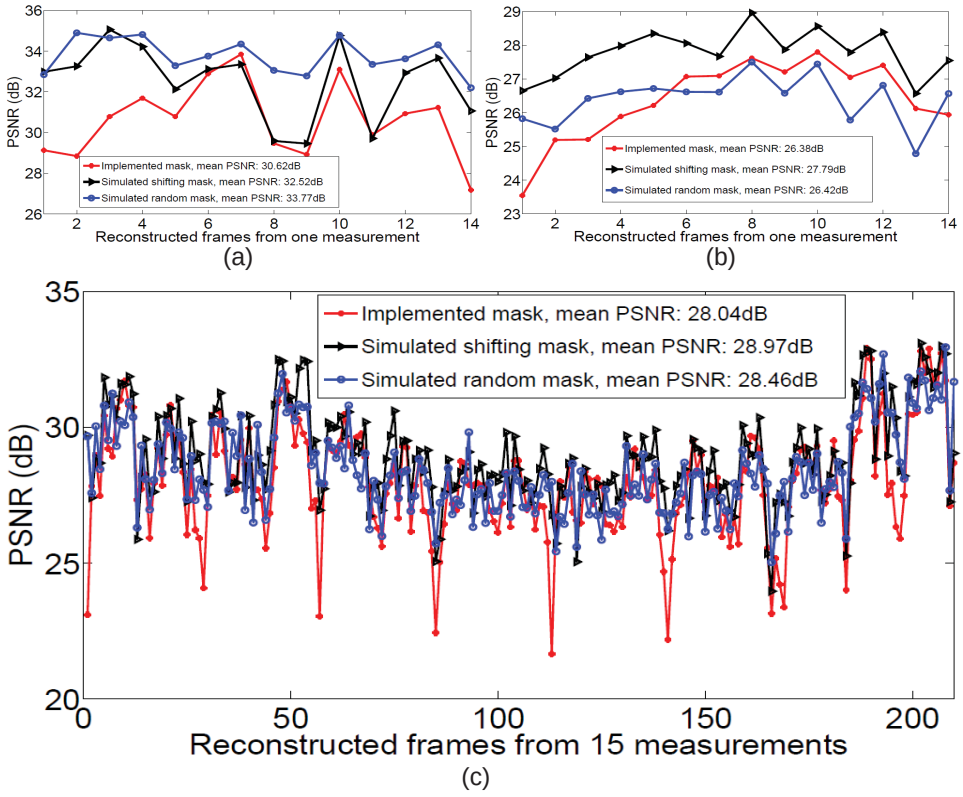


Fig. 15. Simulated and actual reconstruction PSNR by frame. (a), (b), and (c) show PSNR by high-speed, reconstructed video frame for 14 eye blink frames (Media 9), 14 chopper wheel frames (Media 10), and 210 chopper wheel frames (Media 11, Media 12, Media 13, Media 14, Media 15) respectively, from snapshots **g**. Implemented mechanically-translated masks (red curves), simulated translating masks (black curves), and simulated LCoS coding (blue curves) were applied to high-speed ground truth data and reconstructed using GAP. Reconstructions using the implemented mask have a PSNR that periodically drops every 14^{th} reconstructed frame due to the mechanical deceleration time of the piezoelectric stage holding the mask; these frames correspond to the time when the mask changes direction.

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013

of videos in Figs. 15(a),15(b),and 15(c) (see Media 9-11 for the complete videos). For these simulations, reconstructed experimental frames at $N_F = 14$ were summed to emulate a time-integrated image. The high-speed reconstructed frames were used as ground truth. We reapply: 1.) the actual mask pattern; 2.) a simulated CACTI mask moving with motion s_k; and 3.) a simulated, re-randomized coding pattern to each high-speed frame used as ground truth. The reconstruction performance difference between translating the same code and re-randomized the code for each of the N_F reconstructed frames is typically within 1dB.

One important exception arises when an object travels with the same velocity as the shift of the mask. Since the forward matrix's null space structure depends on the spatial structure of the shifting mask pattern, frequencies of f matched with those of T produce reconstruction artifacts arising from structured aliasing. In this unlikely case, the shifting mask compression strategy yields poorer spatial reconstruction quality than that obtainable by re-randomizing the mask at each temporal channel of interest. This result may be overcome in future implementations consisting of two-dimensional or adaptive mask motion.

6. Discussion and conclusion

CACTI presents a new framework to uniquely code and decompress high-speed video exploiting conventional sensors with limited bandwidth. This approach benefits from mechanical simplicity, large compression ratios, inexpensive scalability, and extensibility to other frameworks.

We have demonstrated GAP, a new reconstruction algorithm that can use one of several bases to compactly represent a sparse signal. This fast-converging algorithm requires no prior knowledge of the target scene and will scale easily to lager image sizes and compression ratios. This algorithm was used for all reconstructions except Figs. 12 and 13.

Despite GAP's computational efficiency, large-scale CACTI implementations will seek to minimize the data *reconstructed* in addition to that transmitted to sufficiently represent the optical datastream. Future work will adapt the compression ratio C such that the resulting reconstructed video requires the fewest number of computations to depict the motion of the scene with high quality.

Since coded apertures are passive elements, extending the CACTI framework onto larger values of N only requires use of a larger mask and a greater detector sensing area, making it a viable choice for large-scale compressive video implementations. As N increases, LCoS-driven temporal compression strategies must modulate N pixels C times per integration. Conversely, translating a passive transmissive element attains C times temporal resolution without utilizing any additional bandwidth relative to conventional low-framerate capture.

Future large-scale imaging systems may employ CACTI's inexpensive coding strategy in conjunction with higher-dimensional imaging modalities, including spectral compressive video. Integrating CACTI with the current CASSI system [20] should provide preliminary reconstructions depicting 4-dimensional datasets $\mathbf{f}(x, y, \lambda, t)$.

Acknowledgment

This work was supported by the Knowledge Enhanced Compressive Measurement Program at the Defense Advanced Research Projects Agency, grant N66001–11–1–4002.

#184177 - $15.00 USD Received 25 Jan 2013; revised 26 Mar 2013; accepted 27 Mar 2013; published 23 Apr 2013