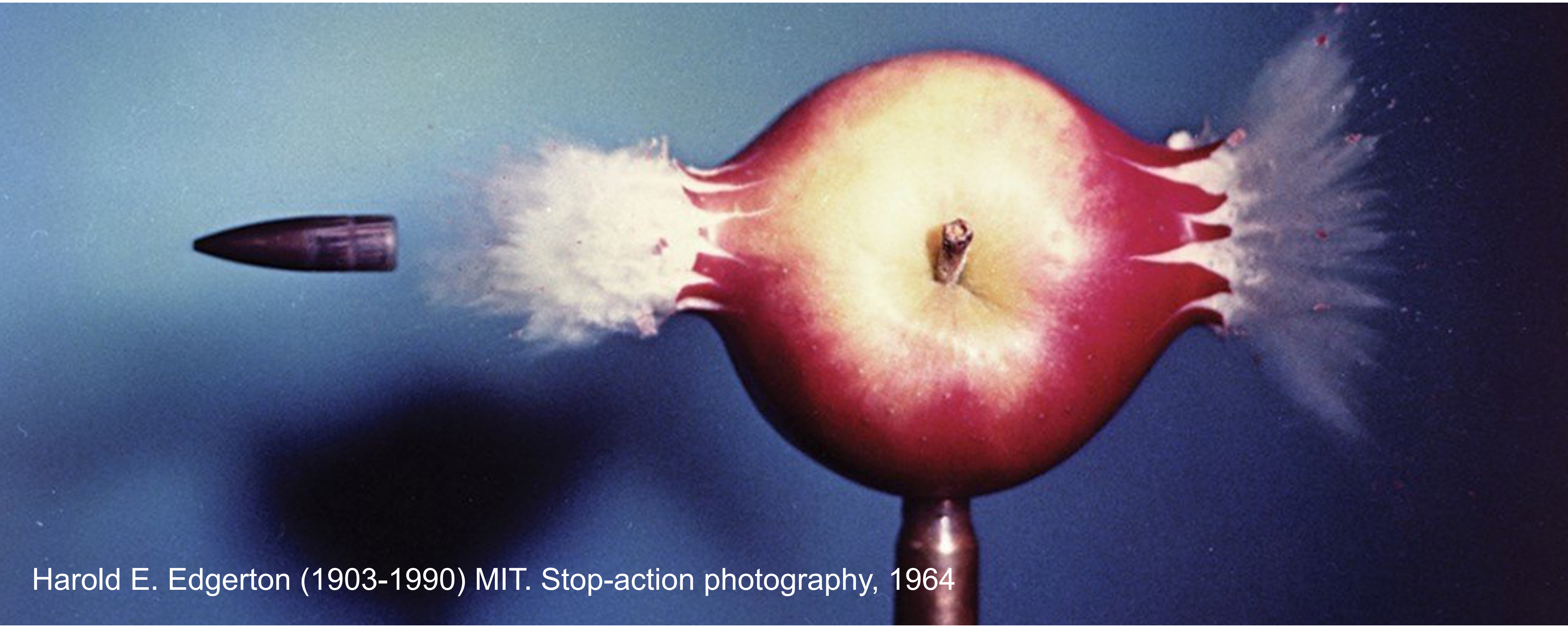
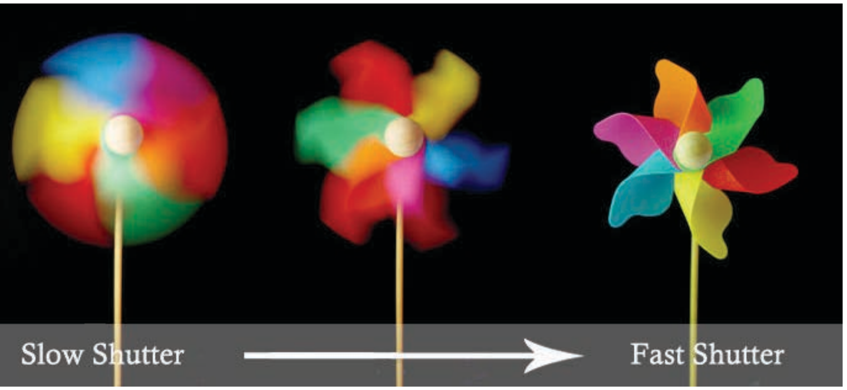
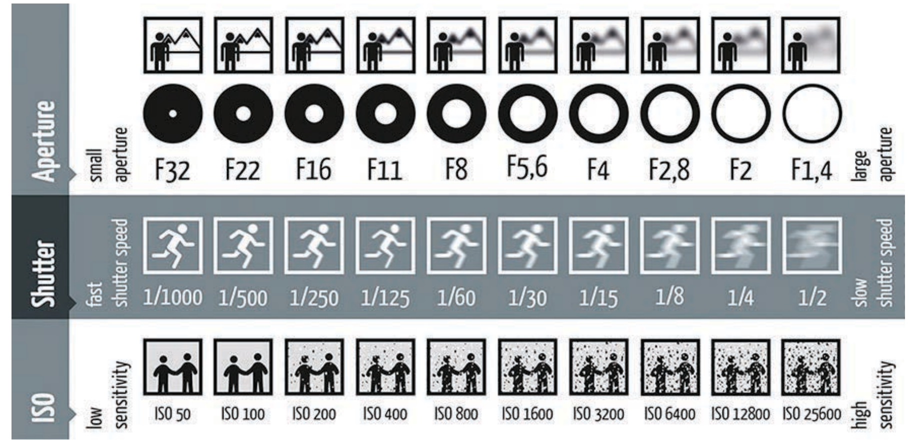
Visualizing Time & Motion



- Capturing time
- Camera Shutter effects, distortions
- Freezing time, blurring image
-

Shutter Speed in Cameras

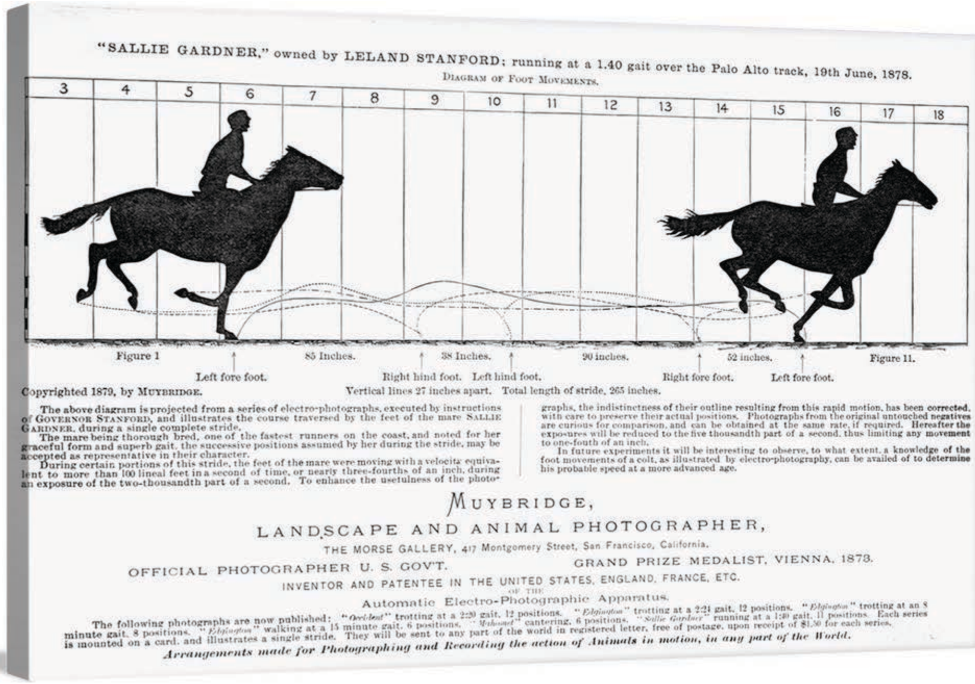


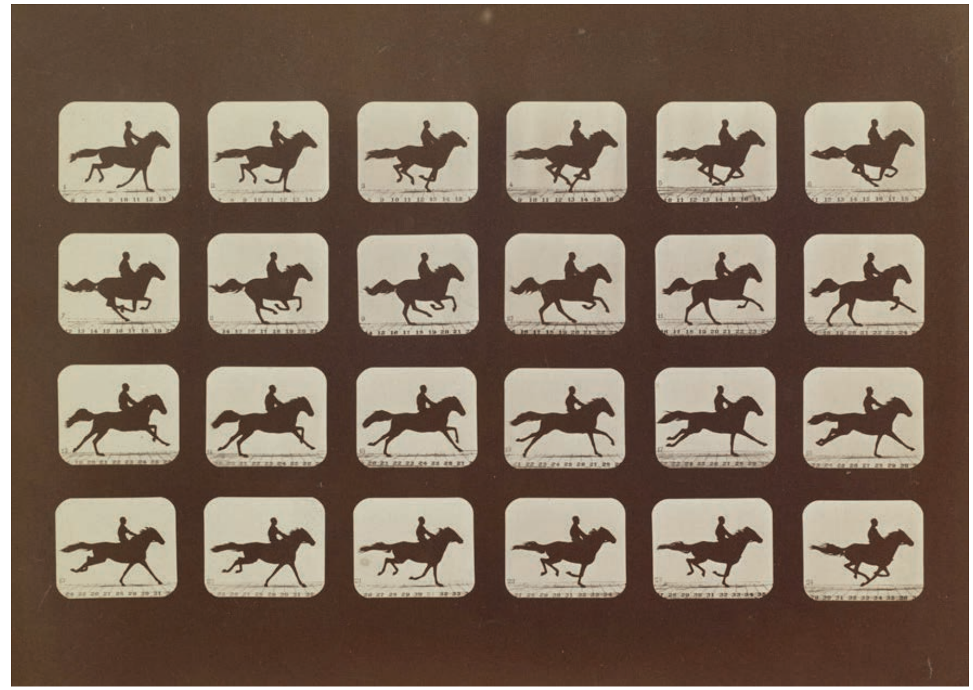


Jacques Henri Lartigue (1912) ICA camera 4x5 with Focal Plane Shutter

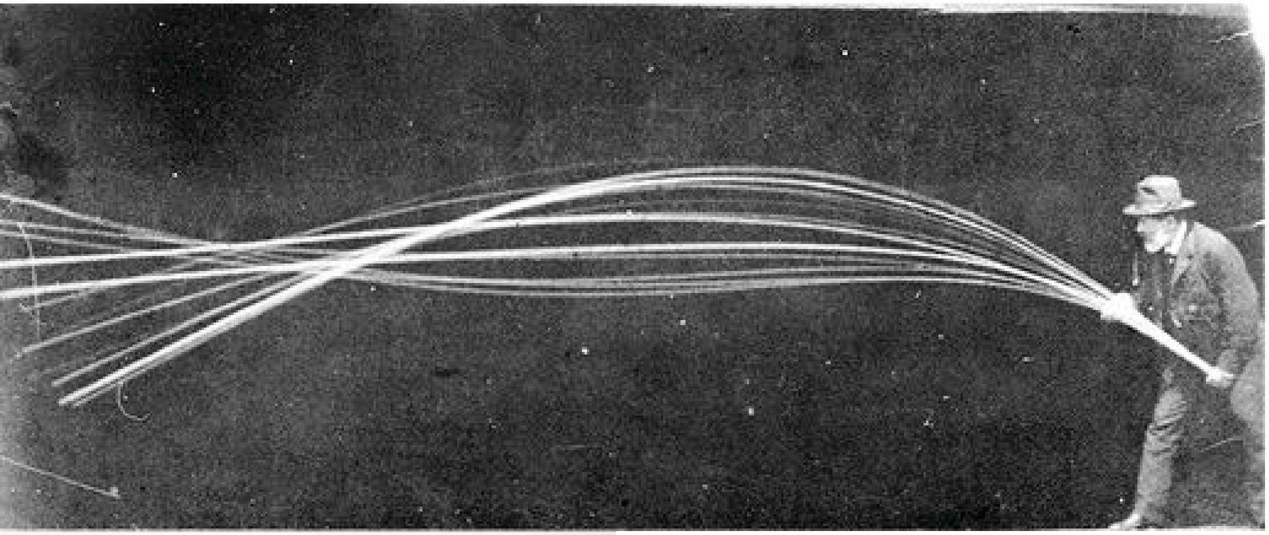
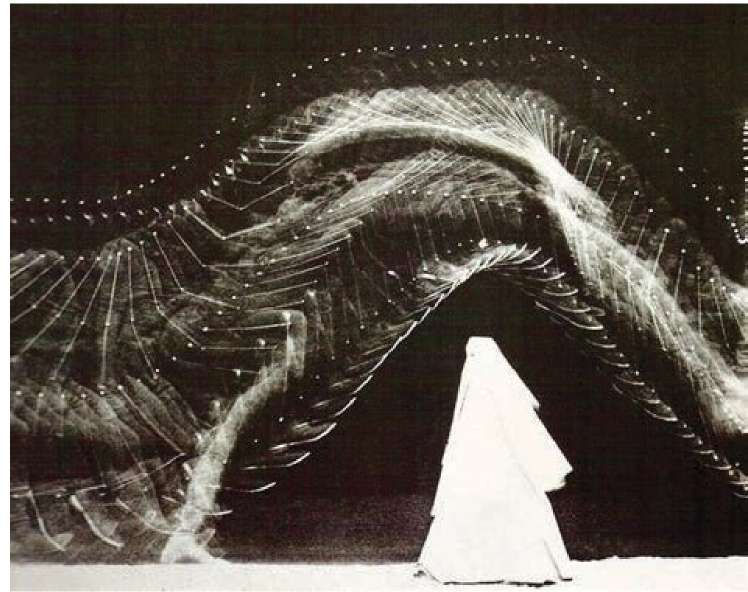
-

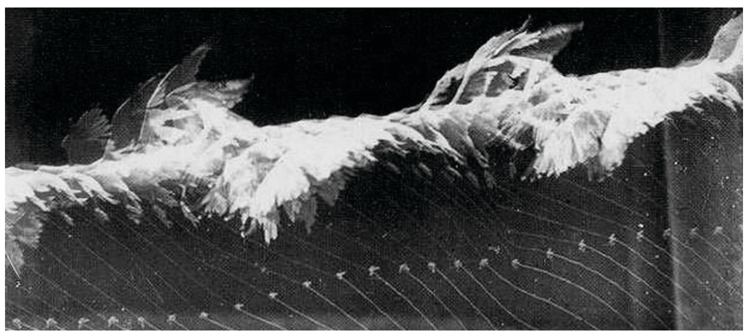
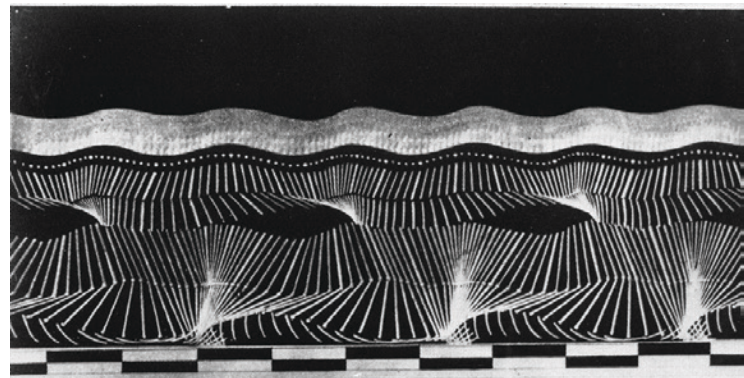
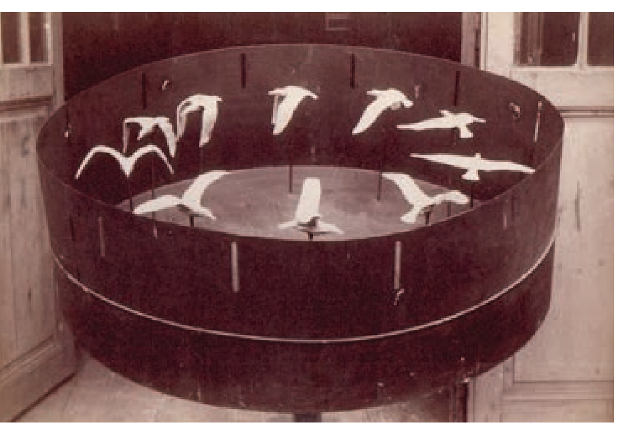
Eadweard Muybridge (1830-1904) Studies in motion

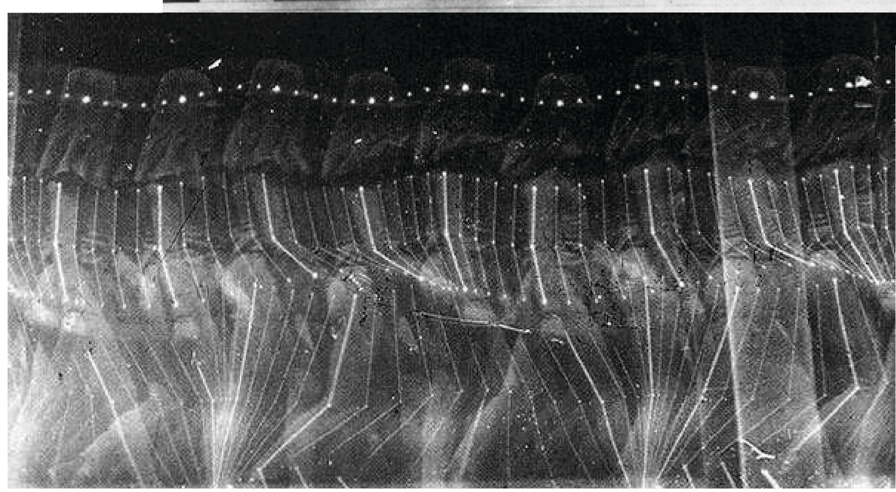




Etienne-Jules Marey (Scientist, chronophotographer, 1880s)



Marcel Duchamp *Nude Descending a Staircase (1912)* | Gerhardt Richter (1965)

Italian Futurism (1909-1930s)



Anton Bragaglia (1911-1913)

Giacomo Balla (1912)

Carlo Carra (1910-1911)

Umberto Boccioni (1913)

Shape-Time Photography

William T. Freeman*
EECS Dept.
Massachusetts Institute of Technology
Cambridge, MA 02139

Hao Zhang*
EECS Dept.
U.C. Berkeley
Berkeley, CA 94720

IEEE Computer Vision and Pattern Recognition (CVPR), Madison, WI, June, 2003.

We introduce a new method to describe shape relationships over time in a photograph. We acquire both range and image information in a sequence of frames using a stationary stereo camera. From the pictures taken, we compute a composite image consisting of the pixels from the surfaces closest to the camera over all the time frames. Through occlusion cues, this composite reveals 3-D relationships between the shapes at different times. We call the composite a shape-time photograph.

Small errors in stereo depth measurements can create artifacts in the shape-time images. We correct most of these using a Markov network to estimate the most probable front-surface pixel, taking into account (a) the stereo depth measurements and their uncertainties, and (b) spatial continuity assumptions for the time-frame assignments of the front-surface pixels.

1 Introduction

With a single still image, we seek to describe the changes in the shape of an object over time. Applications could include artistic photographs, instructional images (e.g., how does the hand move while sewing?), action summarization, and photography of physical phenomena.

How might one convey, in a still image, changes in shape? A photograph depicts the object, of course, but not its relationship to objects at other times. Multiple-exposure techniques, pioneered in the late 1800's by Marey and Murbridge [1, 9] can give beautiful depictions of objects over time. They have two drawbacks, however: (1) The control of image contrast is a problem; the image becomes overexposed where objects at different times overlap. Backgrounds may need to be dark to avoid over-exposure. (2) The result doesn't show how the various shapes relate to each other in three-dimensions. What we see is like an X-ray photograph, showing only a flattened comparision between 2-d shapes.

Using background stabilization techniques from computer vision, researchers have developed video summarization tools which improve on multiple-exposure methods. Researchers at both Sarnoff Labs [13] and Salient Stills [7] have shown single-frame composites where the foreground image at each time overwrites the overlapping portions of all the previous foreground images, over a single, stabilized background. We will refer to this compositing as the "layer-by-time"algorithm, since it is time, not 3-D shape, which determines object visibility. The layer-by-time method avoids the contrast reduction of multiple exposure techniques, However, since temporal order, not shape, determines the occlusion relationships, this method cannot describe the shape relationships between foreground objects at different times. Video cubism [5] is a less structured approach to rendering video information into a single frame, and also does not incorporate shape information into the composite.

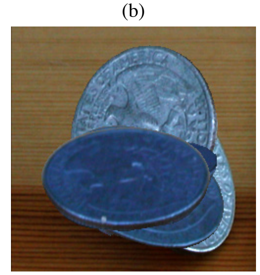
Our solution for displaying shape changes over time makes use of 3-D information which is captured along with the images. We form a composite image where the pixels displayed are those showing the surfaces closest to the viewer among all surfaces seen over the entire sequence. The effect is to display a photograph of the union of the surfaces in all the photographs (without mutual illumination and shading effects). This allows occlusion cues to reveal the 3-D shape relationships between objects seen over different times in the original video sequence.

Figure 1 illustrates these summarization methods for the case of a familiar motion sequence: the rattling spiral of a coin as it rolls to a stop on a table. (a) shows the individual frames of the sequence. (To avoid motion blur, we placed the coin in those positions, using clay underneath). The multiple-exposure summary, (b), shows the loss of image contrast where foreground objects overlap. The layer-by-time algorithm, (c), shows more detail than (b), but doesn't reveal how the coins of different times relate spatially. (d) is our proposed summary of the sequence. The composite

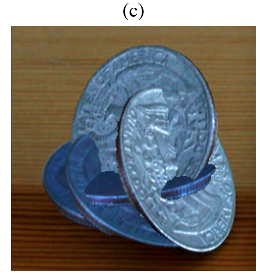
*This work was initiated when both authors were at Mitsubishi Electric Research Labs (MERL), WTF as a researcher and HZ as a student intern.

(a)

(b)



(c)



(d)

Figure 1: (a) Image sequence of rolling coin. (b) Multiple exposure summary. (c) Layer-by-time summary. (d) Shape-time summary. (Color-based foreground masks were used in (c) and (d) to isolate the foreground coins from the background: in (c) to specify the foreground object and in (d) to remove the unreliable stereo depth for the featureless background.)

image is constructed to make sense in 3-D. We can see how the coin occludes itself at other times; these occlusions let us picture the 3-D relationships between the different spatial configurations of the coin. To emphasize that the technique describes shapes over time, we call it "shape-time photography".

1.1 Related effects

In some special cases of natural viewing, we are accustomed to viewing shape-time images. Extrusion processes, such as squeezed blobs of toothpaste or shaving cream, leave a shape-time history of the motion of the extrusion source. Shape-time photographs have some resemblance to Duchamp's "Nude Descending a Staircase", the classic depiction of motion and shape in a static image. The comic book Nogenon uses drawn shape-time outlines in its story [14]. In unpublished independent work, researchers at Georgia Tech have made graphical displays of data from a motion-capture system using a shape-time style rendering, but not using visual input [2].

2 Problem Specification

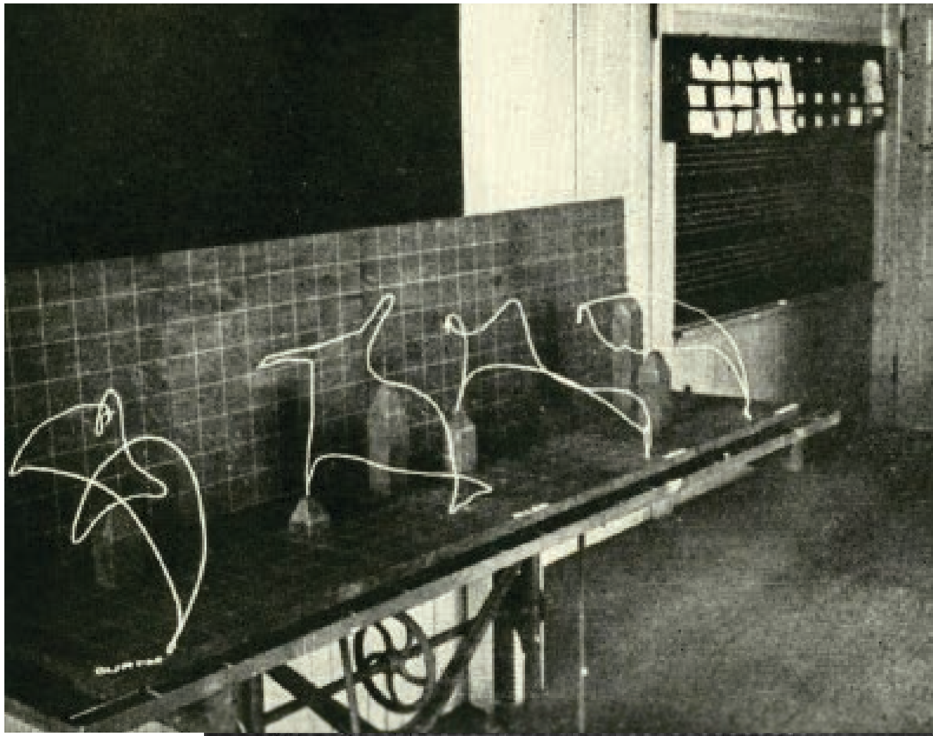
To make a shape-time photograph, we need to record both image and depth information. Various technologies can measure depth everywhere in a scene, including shape-from-defocus, structured light systems, and stereo. While stereo range can be less accurate than others, a stereo camera is quite portable, allowing a broad range of photographic subjects in different locations. Stereo also avoids the problem of registering range and image data, since disparities are computed from the image data itself. Fig. 2 shows the stereo camera we used. The beam-splitter system allowed us to capture left and right images using a single shutter, assuring temporally synchronized images.

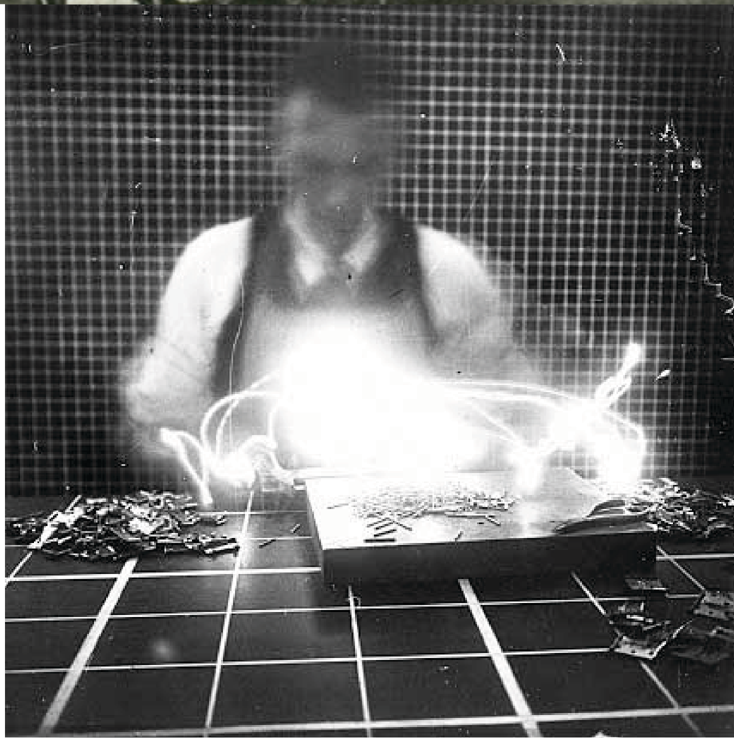
The simplest version of shape-time photography assumes a stationary camera which photographs N time-frames of stereo image pairs. (Background stabilization techniques such as [16] might be used to generalize the results of this paper to non-stationary cameras). At each position, we need to select for display a pixel from one of the N frames captured over all times at that position. We can then generate a single-frame composite, from one camera's viewpoint (left, for our examples), or a composite stereo image.

Let $L_k(t)$ and $R_k(t)$ denote the values at the kth pixel at time frame t recorded in the left and right images, respectively. Let $d_k(t)$ be the distance to the surface imaged at the kth pixel (of the left camera) at frame t. Pixel k of the left view shape-time image, I, is simply

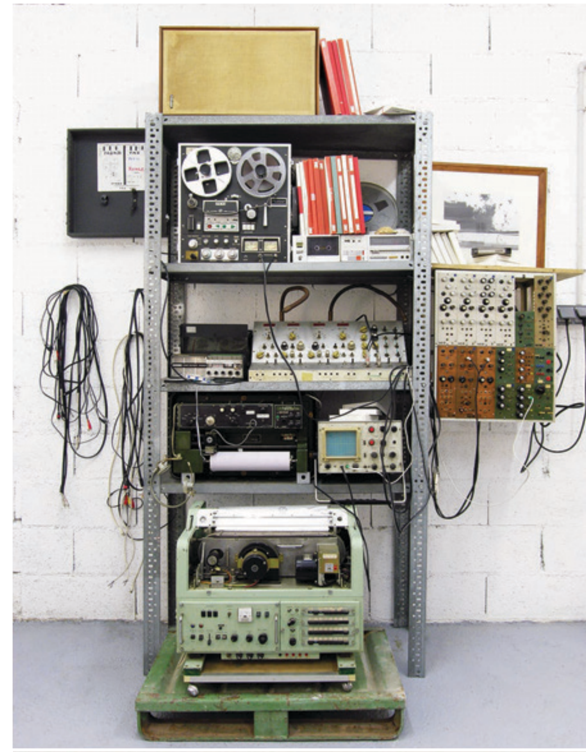
$$I_k = L_k(\text{argmin}_t d_k(t)) \quad (1)$$

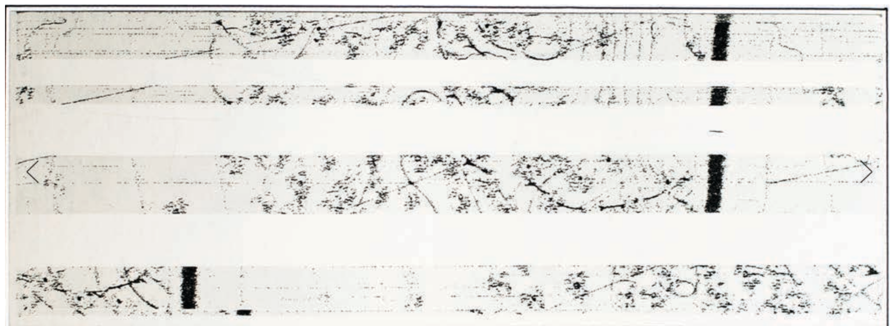
Frank & Lillian Gilbreth Time-Motion Studies (1914)

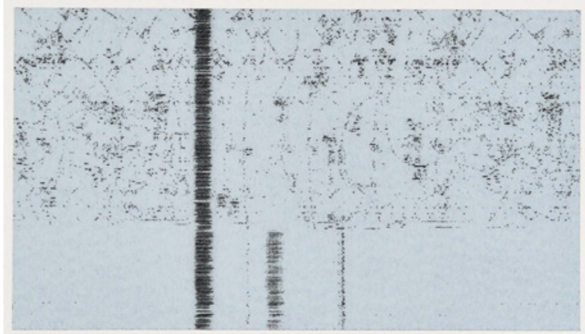


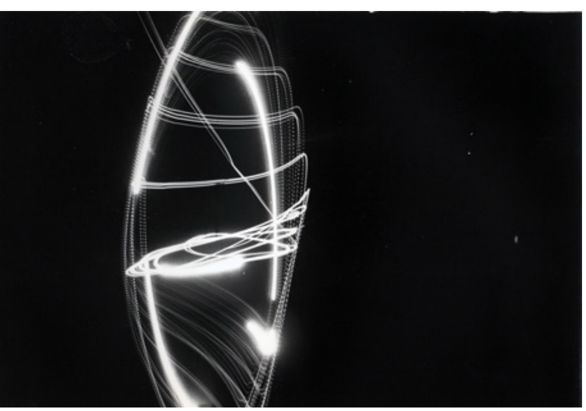


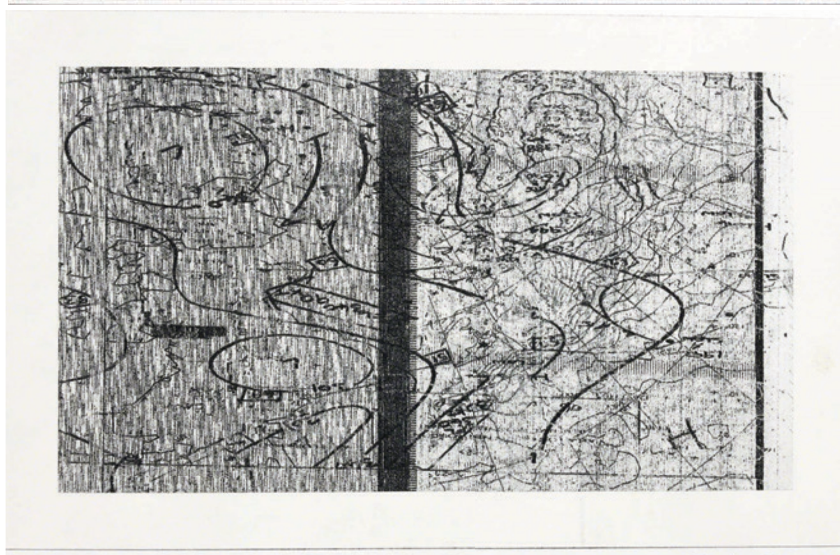
Lars Fredrickson (1926-1997) Multi-disciplinary artist

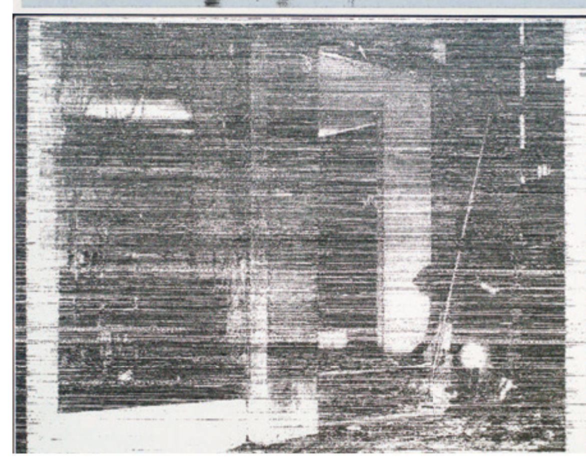




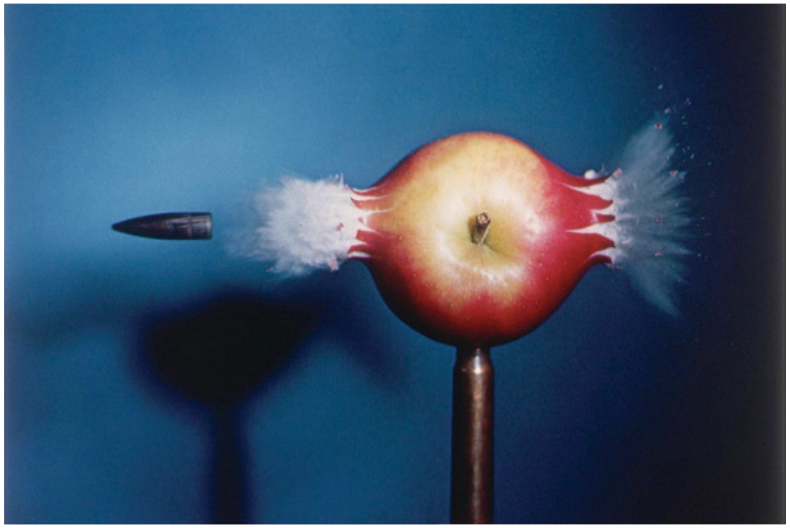
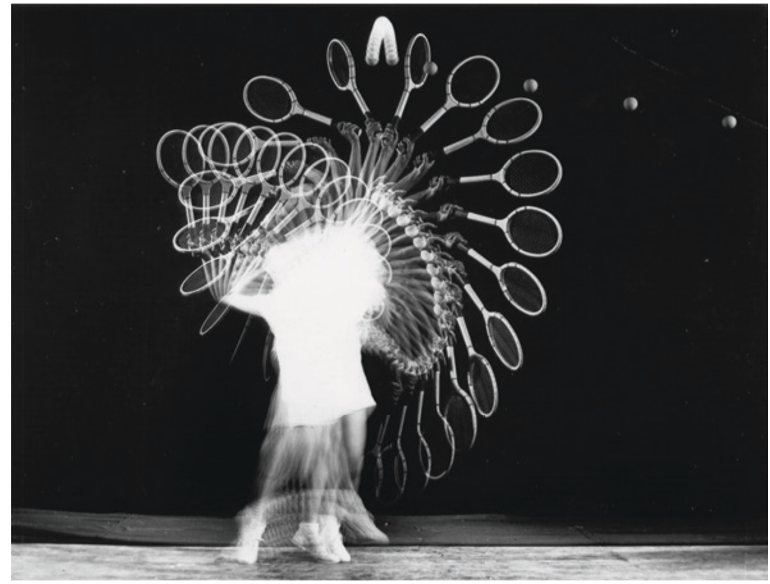
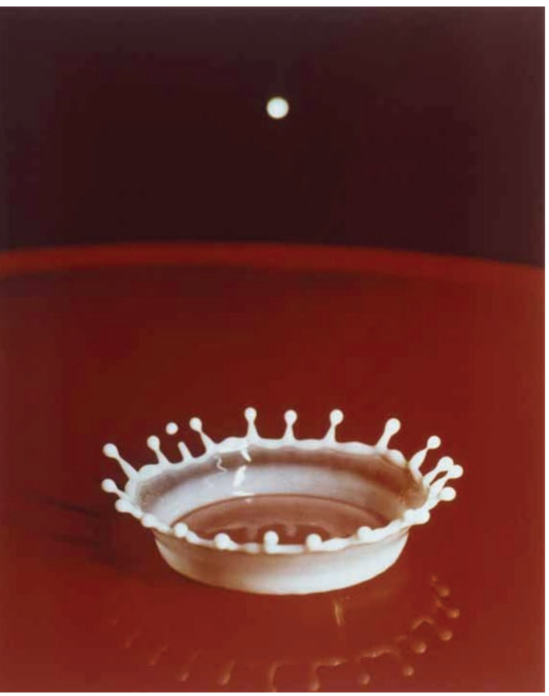
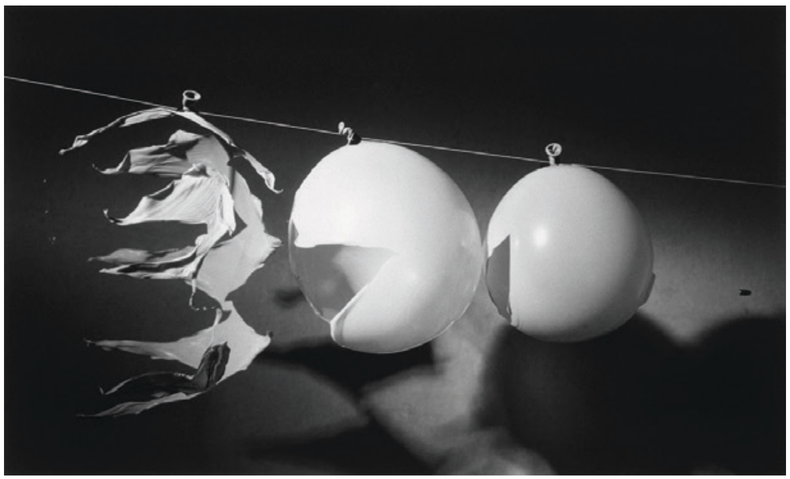
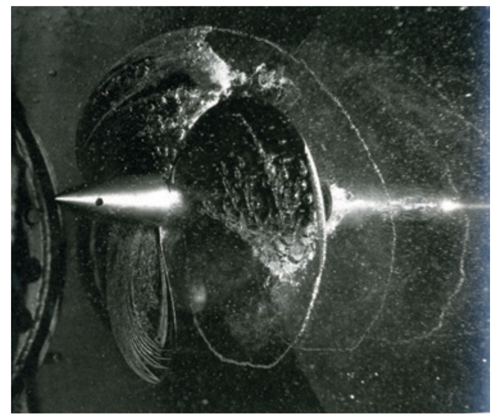
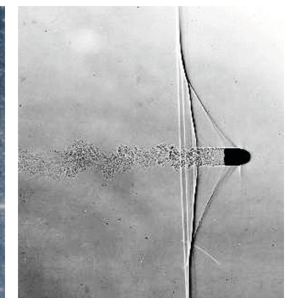


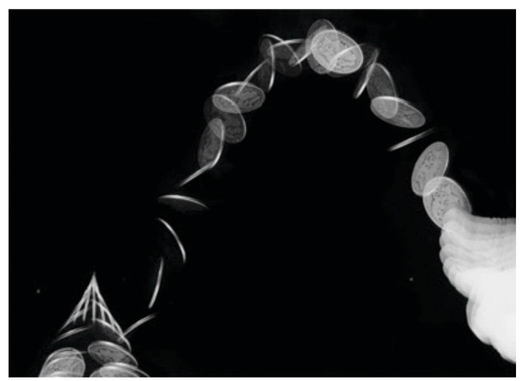
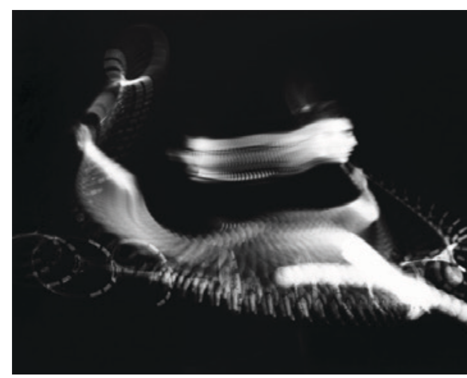






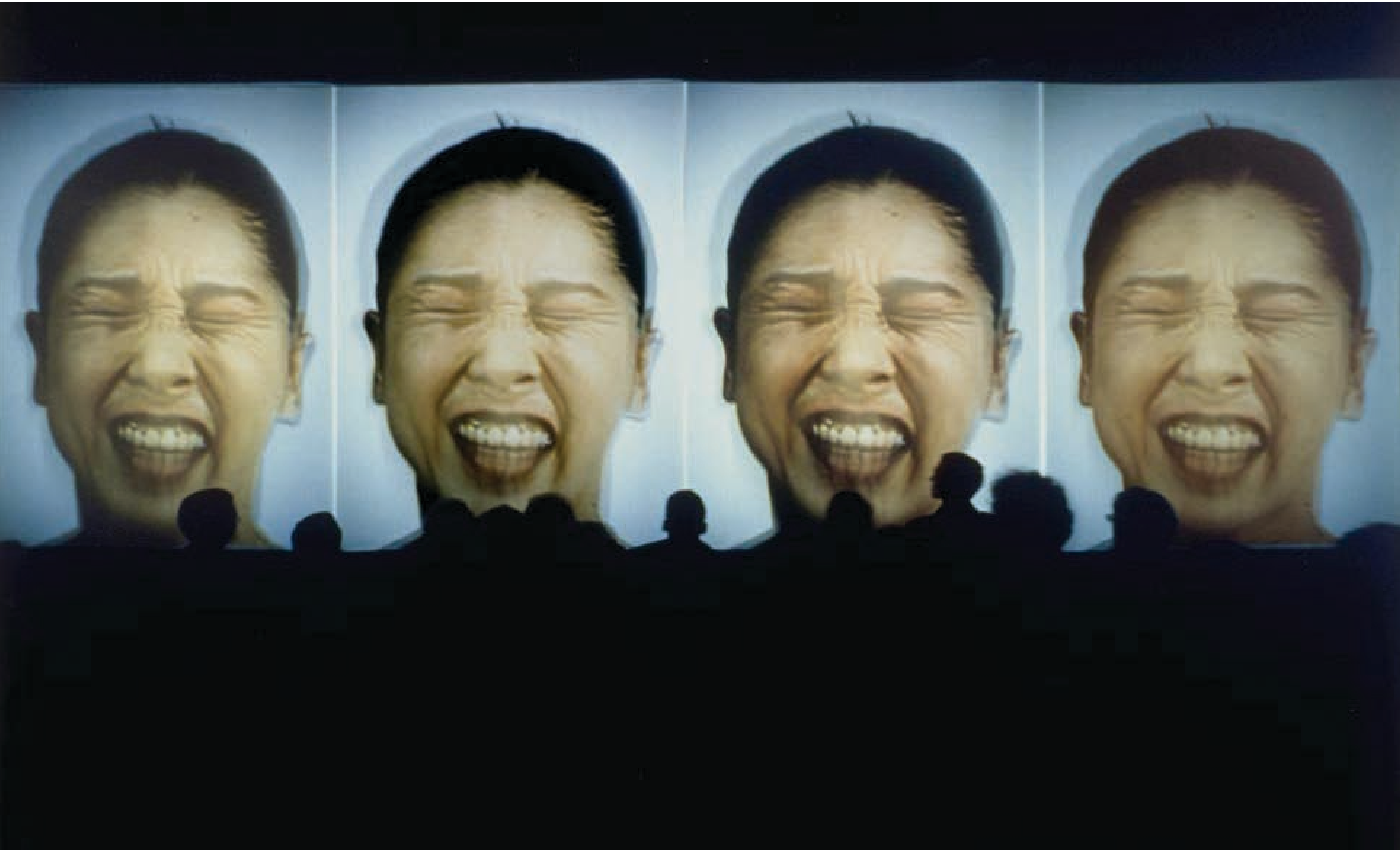
Harold E. Edgerton, Strobe-photography, MIT (1903-1990)

Granular Synthesis <u>Modell 5</u> (Akemi Takeya) 1994-1996

“The idea was to create a media figure oscillating between ”naturalness and artificially” one that could be both seductive and violent, both desperate and robotic, a Cyborg, an attractive/repulsive, alien/familiar hermaphrodite”



Coded Exposure Photography: Motion Deblurring using Fluttered Shutter

Ramesh Raskar* Amit Agrawal
Mitsubishi Electric Research Labs (MERL), Cambridge, MA

Jack Tumblin
Northwestern University

(a) Blurred Image (b) Rectified Crop (c) Deblurred Image

Figure 1: Coded exposure enables recovery of fine details in the deblurred image. (a) Photo of a fast moving vehicle. (b) User clicks on four points to rectify the motion lines and specifies a rough crop. (c) Deblurred result. Note that all sharp features on the vehicle (such as text) have been recovered.

Abstract

In a conventional single-exposure photograph, moving objects or moving cameras cause motion blur. The exposure time defines a temporal box filter that smears the moving object across the image by convolution. This box filter destroys important high-frequency spatial details so that deblurring via deconvolution becomes an ill-posed problem.

Rather than leaving the shutter open for the entire exposure duration, we "flutter" the camera's shutter open and closed during the chosen exposure time with a binary pseudo-random sequence. The flutter changes the box filter to a broad-band filter that preserves high-frequency spatial details in the blurred image and the corresponding deconvolution becomes a well-posed problem. We demonstrate that manually-specified point spread functions are sufficient for several challenging cases of motion-blur removal including extremely large motions, textured backgrounds and partial occluders.

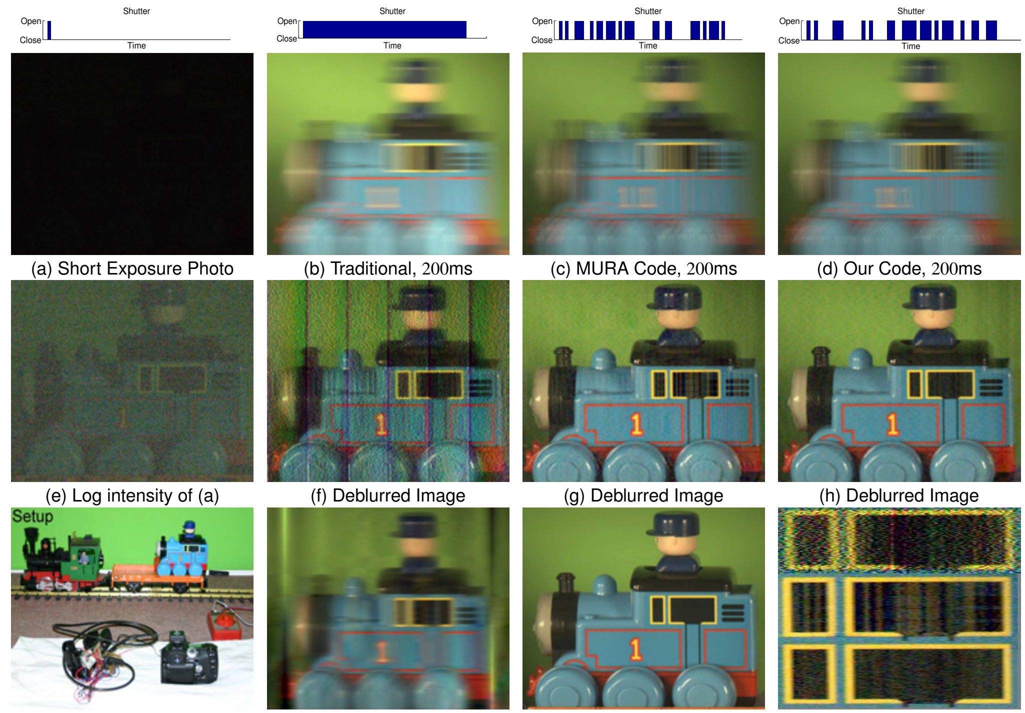
1. Introduction

Despite its usefulness to human viewers, motion is often the bane of photography: the clearest, most detailed digital photo requires a perfectly stationary camera and a motionless scene. Relative motion causes motion blur in the photo. Current practice presumes a 0^{th} order model of motion; it seeks the longest possible exposure time for which moving objects will still appear motionless. Our goal is to address a first-order motion model: movements with constant speed rather than constant position. Ideally, the camera would enable us to obtain a sharp, detailed record of each moving component of an image, plus its movement.

This paper takes first steps towards this goal by recoverably encoding large, first-order motion in a single photograph. We rapidly open and close the shutter using a pseudo-random binary sequence during the exposure time so that the motion blur itself retains decodable details of the moving object. This greatly simplifies the corresponding image deblurring process. Our method is not fully automatic: users must specify the motion by roughly outlining this modified blurred region. We then use deconvolution to compute sharp images of both the moving and stationary components within it, even those with occlusions and linear mixing with the background.

Deconvolution to remove conventional motion blur is an old, well-explored idea, but results are often disappointing. Motion blurred images can be restored up to lost spatial frequencies by image deconvolution [Jansson 1997], provided that the motion is shift-invariant, at least locally, and that the blur function (point spread function, or PSF) that caused the blur is known. However, image deconvolution belongs to the class of ill-posed inverse problems for which the uniqueness of the solution cannot be established, and the solutions are oversensitive to any input data perturbations [Hadamard 1923] [Tikhonov and Arsenin 1977]. In comparison, the proposed modification of the capture process makes the deblurring problem well-posed.

*e-mails: [raskar,agrawal]@merl.com, jet@cs.northwestern.edu.
Web: http://www.merl.com/people/raskar/deblur



(a) Short Exposure Photo (b) Traditional, 200ms (c) MURA Code, 200ms (d) Our Code, 200ms
(e) Log intensity of (a) (f) Deblurred Image (g) Deblurred Image (h) Deblurred Image
(i) Toy Train Setup (j) RL deblur result of (b) (k) Photo of Static Toy Train (l) Deblurring details

Figure 6: Comparison with other exposure settings: short exposure, traditional shutter exposure, MURA code and our code. The blur k in each case is between 118 and 121 pixels ($\approx 16\%$ of n). (a,b,c,d) Shutter sequence and corresponding photos used as input images. (e) Log intensity for short exposure. (f,g,h) Deblurred results using a linear solution. (i) Experimental setup with toy train. (j) Flat blurred image deblurred using Richardson-Lucy (RL) algorithm. (k) Photo of static toy train. (l) Enlarged regions taken from deblurred results for flat (top), MURA (middle) and coded exposure (bottom). Datasets and source code available at `http://www.merl.com/people/raskar/deblur/`.

settled for a compromise value by experimentation, choosing a sequence of $m = 52$ chops with 50% duty cycle, i.e., with 26 ones and zeros. The first and last bit of the code should be 1, which results in ${}^{50}C_{24} \approx 1.2 \times 10^{14}$ choices. Among them, there are a multitude of potential candidates with acceptable frequency magnitude profile but different phase. We computed a **near-optimal code** by implementing a randomized linear search and considered approximately 3×10^6 candidate codes. We chose a code that (i) maximizes the minimum of the magnitude of the DFT values and (ii) minimizes the variance of the DFT values. The near-optimal code we found is

1010000111000001010000110011110111010111001001100111.

The plot in Figure 5 demonstrates that the chosen code is a significant improvement over padded MURA code. The deblurred images in Figure 6 shows banding and artifacts for flat blur and MURA coded blur as compared to coded blur using our code.

4. Motion Decoding

Given the estimated PSF, we can deblur the captured high resolution image using existing image deconvolution algorithms. However, in several cases described below, we discovered that adding more constraints is difficult via deconvolution, and instead a linear algebra approach is more practical.

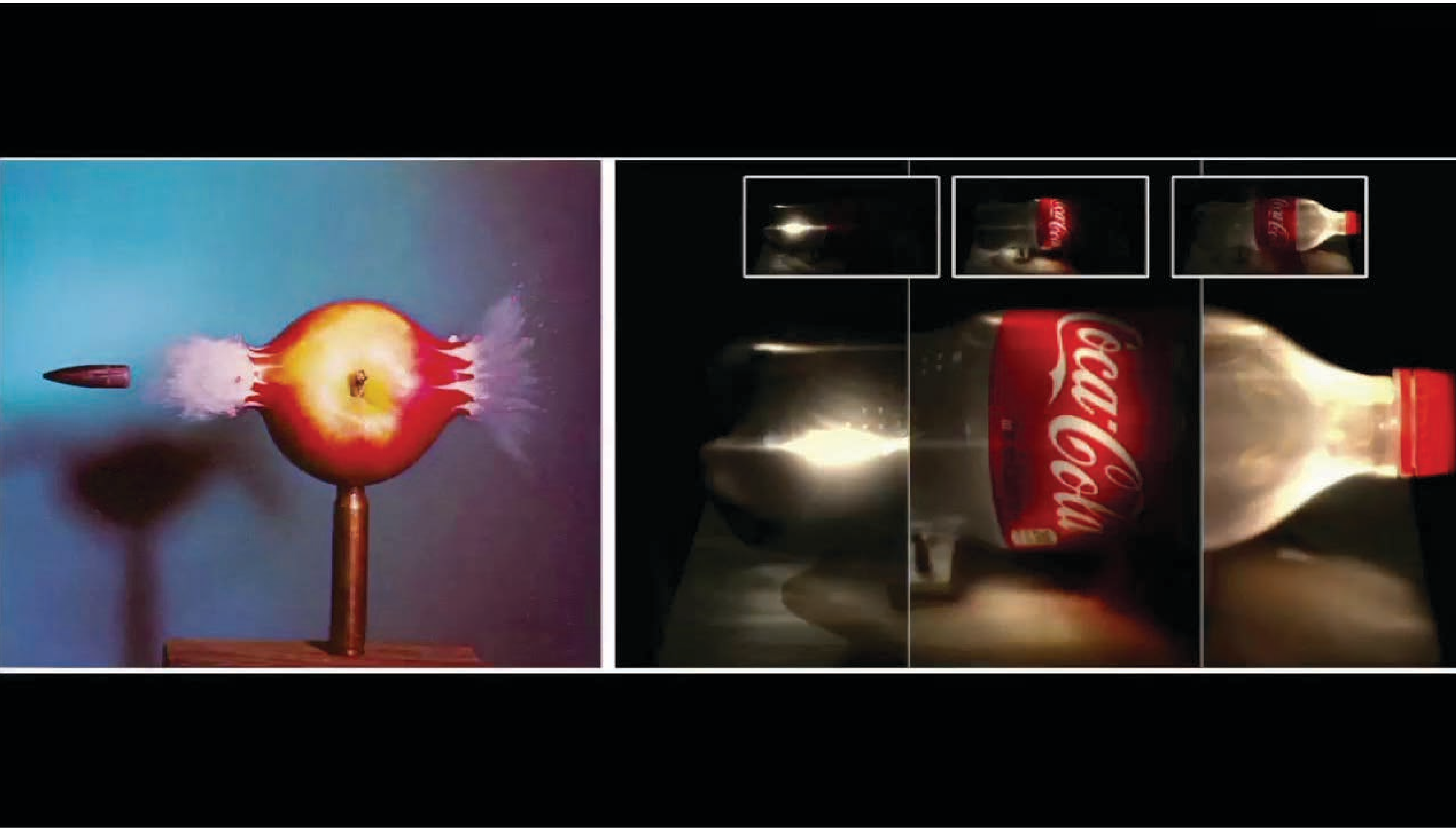
4.1. Linear Solution

We use a least-square estimation to solve for the deblurred image $\widehat{X}$ as

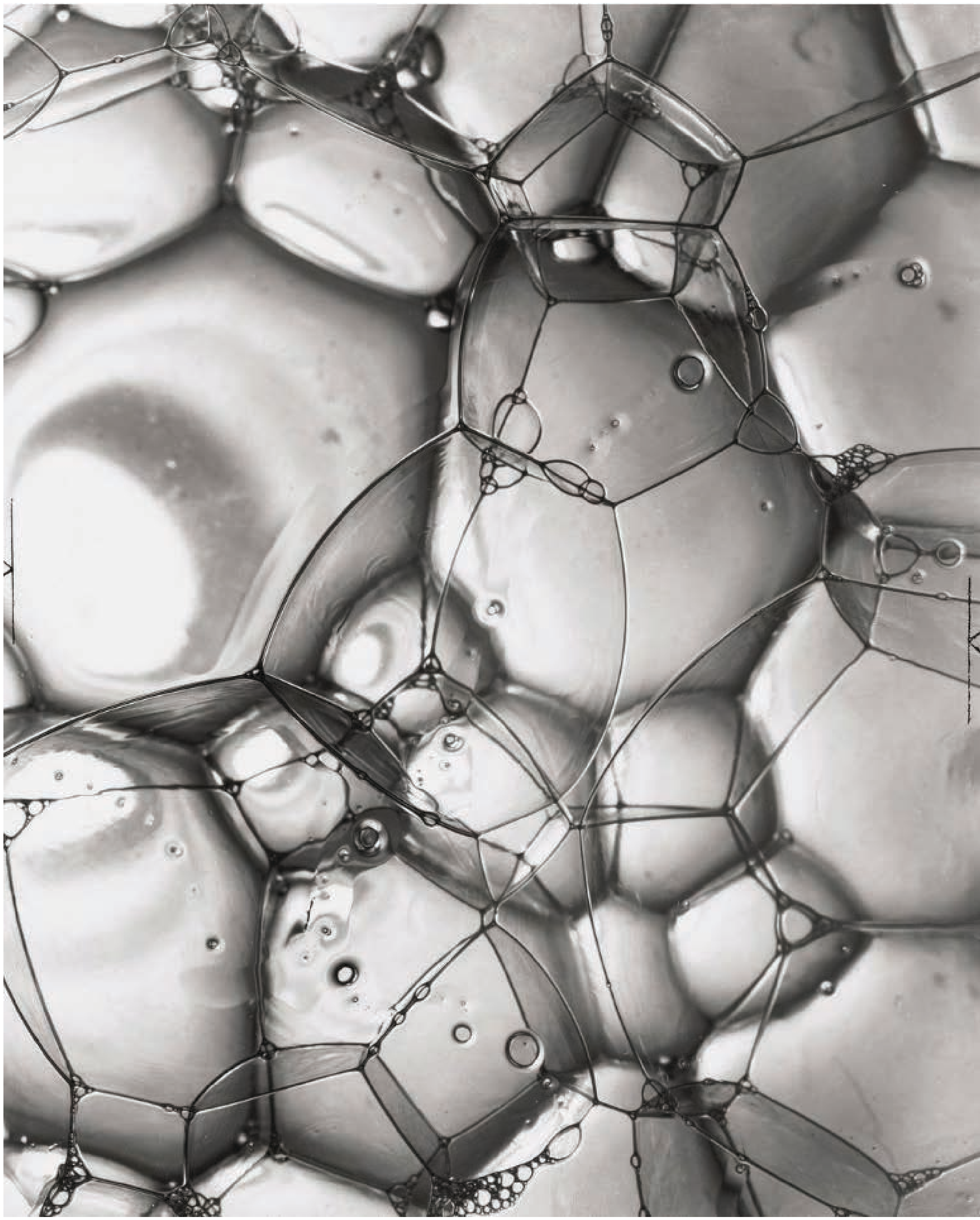
$$\widehat{X} = A^{+}B, \tag{4}$$

where A^{+} is the pseudo-inverse of A in the least-square sense. Since the input image can have a motion blur k different from m, we first expand/shrink the given blurred image by factor m/k. We then estimate X and scale it back by k/m. All the images in this paper have been deblurred using this simple linear approach with no additional post-processing.

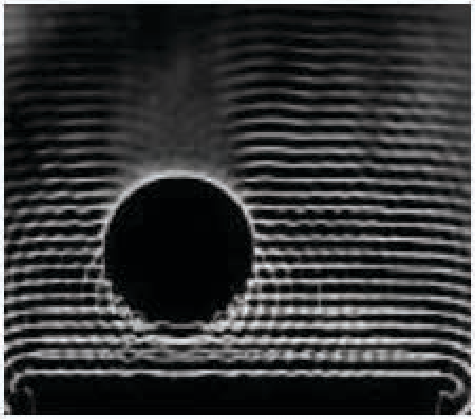
In the following sections, we focus on one dimensional PSFs. Motion of real-world objects within a frame tends to be one dimensional due to energy and inertial constraints. We refer to the one dimensional line-like paths for motion as *motion lines*. Note that scene features on a given motion line contribute only to pixels on that motion line and therefore the motion lines are independent. The solution for each motion line can be computed independent of other motion lines. In the explanation below, without loss of generality, Ramesh Raskar MIT Media Lab Camera Culture Femto Photography

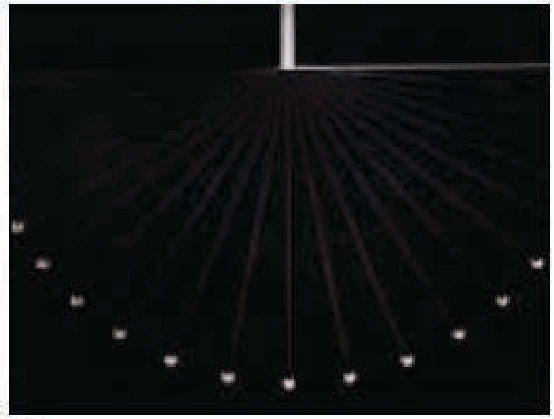
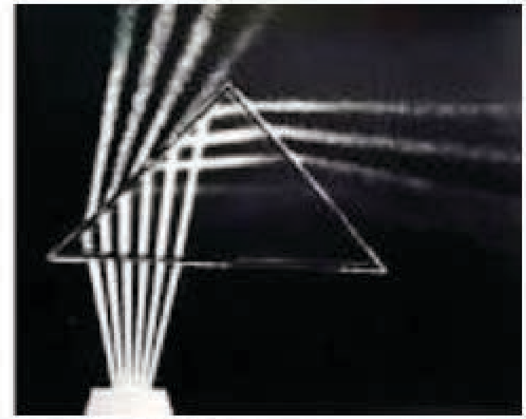
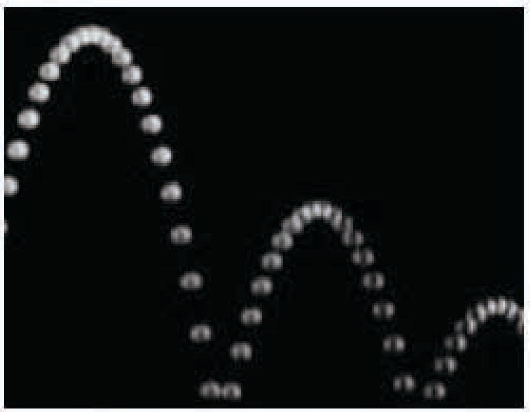
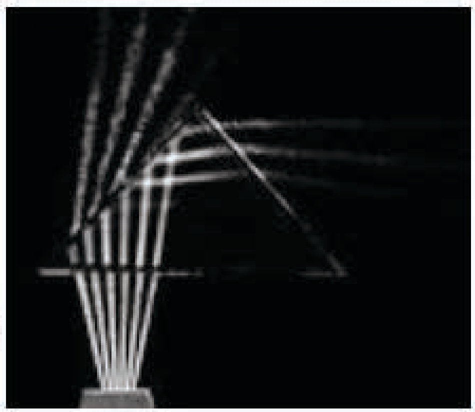
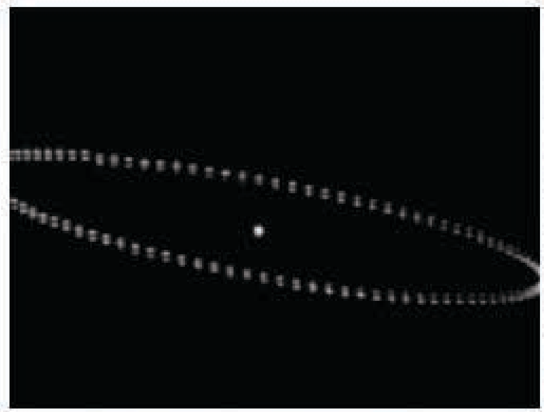
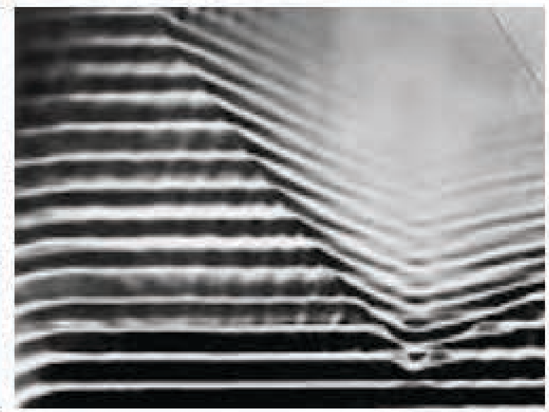


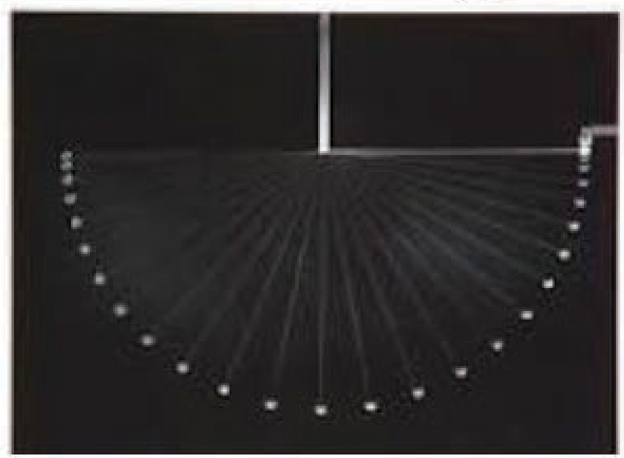
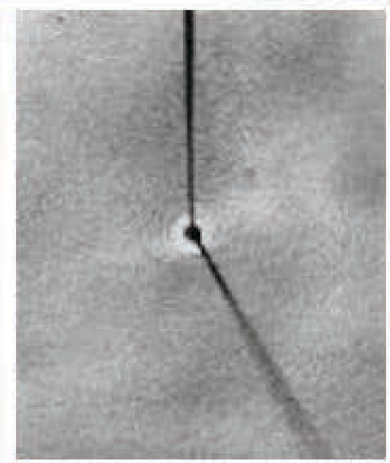
Berenice Abbott (1898-1991) *Soap Bubbles* (1946)



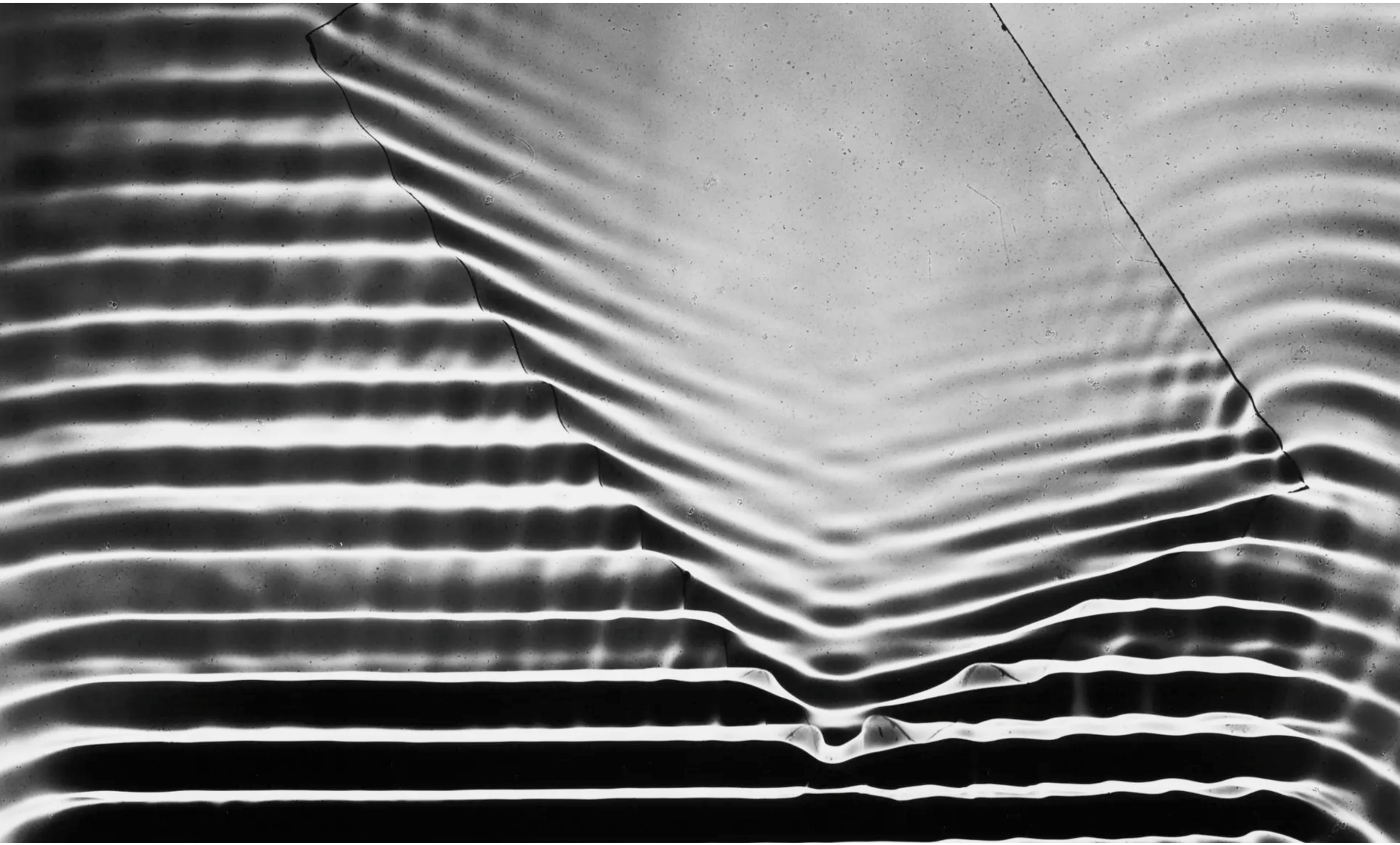
Berenice Abbott *Wave Patterns*



Berenice Abbott ***Wave Pattern***



Ruth Jarman, Joe Gerhardt Semiconductor (2011)