A method for the real-time rendering of formless dot field structure-from-motion stimuli

Jedediah M. Singer
Brain Science Program, Brown University,
Providence, RI, USA

David L. Sheinberg
Department of Neuroscience, Brown University,
Providence, RI, USA

The perception of visual motion relies on different computations and different neural substrates than the perception of static form. It is therefore useful to have psychophysical stimuli that carry mostly or entirely motion information, conveying little or nothing about form in any single frame. Structure-from-motion stimuli can sometimes achieve this dissociation, with some examples in studies of biological motion using point-light walkers. It is, however, generally not trivial to provide motion information without also providing static form information. The problem becomes more computationally difficult when the structures and the motions in question are complex. Here we present a technique by which an animated three-dimensional scene can be rendered in real-time as a pattern of dots. Each dot follows the trajectory of the underlying object in the animation, but each static frame of the animation appears to be a uniform random field of dots. The resulting stimuli capture motion vectors across arbitrary complex scenes, while providing virtually no instantaneous information about the structure of that scene. We also present the results of a psychophysical experiment demonstrating the efficacy and the limitations of the technique. The ability to create such stimuli on the fly allows for interactive adjustment and control of the stimuli, real-time parametric variations of structure and motion, and the creation of large libraries of actions without the need to pre-render a prohibitive number of movies. This technique provides a powerful tool for the dissociation of complex motion from static form.

Keywords: structure from motion, motion perception, biological motion, animate form

Citation: Singer, J. M., & Sheinberg, D. L. (2008). A method for the real-time rendering of formless dot field structure-from-motion stimuli. *Journal of Vision, 8*(5):8, 1–8, http://journalofvision.org/8/5/8/, doi:10.1167/8.5.8.

Introduction

Many objects in the world move; even motionless objects constantly move relative to our eyes as we navigate and explore the world. Numerous previous investigators have used psychophysical, neuroimaging, or electrophysiological methods to explore how different kinds of motion can support object segmentation and form perception. The appearance of motion can also arise from static form, for example when viewing a flipbook or a cartoon. Our perception of "something moving" is generally unified and coherent, but for experimental purposes it can be very useful to dissociate the "something"—that is, the form—from how it is moving.

Structure-from-motion stimuli, in which objects with ambiguous structure become unambiguous when set into motion, have provided many insights into the relationship between motion and form. Wallach and O'Connell (1953) described the "kinetic depth effect," in which flat projections of wireframes spring into three dimensions when the wireframes are rotated. Later studies have used random dot patterns applied to the surfaces of objects. As the objects move, each dot follows the trajectory of the underlying object at that point, and the collection of dots gives roughly the same motion impression as the original object. Such stimuli contain enough information to judge relative surface slant (Domini & Caudek, 1999), curvature (van Damme & van de Grind, 1996), and form (Doner, Lappin, & Perfetto, 1984). When uniform dot textures are mapped onto the surfaces of objects, it does not necessarily translate to a uniform distribution of dots as seen by the viewer. As a surface slants away from frontoparallel, apparent dot density increases. This can give significant form cues even in the absence of motion (Figure 1). This can be ameliorated by evaluating dot distribution in each frame and adding and removing dots to bring the distribution closer to uniform (as in Sperling & Landy, 1989). However, this increases scintillation at the edges of rotating objects and can be computationally expensive.

Reducing the number of dots can also help to decrease the contribution of static form. One prominent example is the study of perception of biological motion using point-light walkers. These stimuli, first used by Johansson (1973), are created by placing a small number of dots at points on an otherwise invisible moving human body. Still images from the resulting movies often appear to be formless collections of dots but when viewed in motion the actions of the underlying body are strikingly clear. Several experiments have attempted to distinguish

doi: 10.1167/8.5.8 Received July 3, 2007; published May 23, 2008

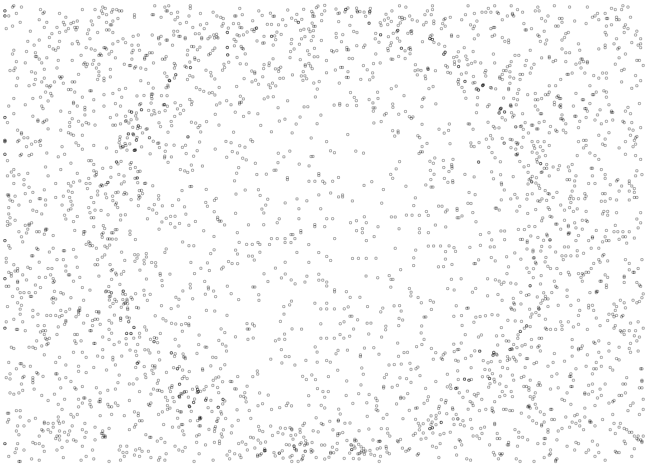


Figure 1. A uniform random dot texture mapped onto the surface of a sphere. Even though only the dots are visible, the variations in dot density caused by surfaces slanted away from the point of view make the underlying sphere apparent.

between motion-based and form-based processing of these stimuli. It is possible that the strong perception of animate motion arises from particular patterns of motion in the stimulus, regardless of the form that is generating that motion (Casile & Giese, 2005; Hoffman & Flinchbaugh, 1982). It may also be that as information accrues over time, the dots in each individual frame obey the location constraints imposed by a particular motion, activating a form-based template of that motion (Beintema & Lappe, 2002). Some arguments have been for a purely form-based or a purely motion-based source for the vivid perception of biological motion; as one recent model (Giese & Poggio, 2003) formalized, both processes could combine to yield the final coherent percept.

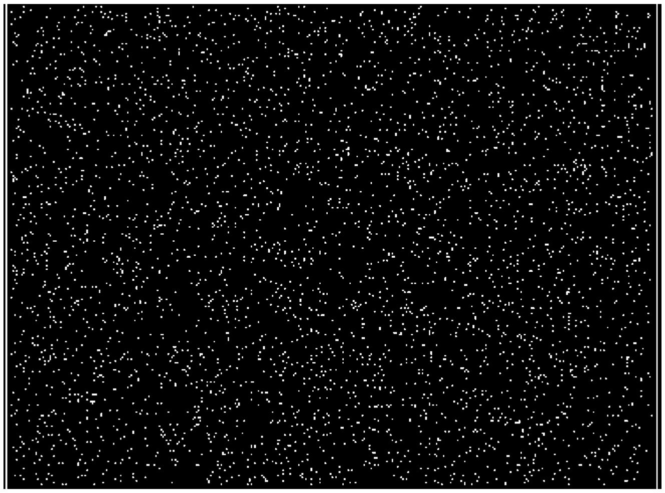
Work outside the field of biological motion has also offered insights into the connections between apparent motion and apparent form. A recent study suggests that low-level motion information can give rise to a form-based representation that then affects how the motion is perceived (Caplovitz & Tse, 2007). Similarly, apparent motion caused by sequential static views only develops after those static views are parsed for form (Tse & Logothetis, 2002). At least one neurological patient is unable to perceive structure from motion, despite being able to recognize static objects and to perceive basic properties of motion (Cowey & Vaina, 2000); another is able to perceive structure from motion, despite severe impairment of low-level motion perception (Vaina, Lemay, Bienfang, Choi, & Nakayama, 1990).

This article describes a technique for the generation of stimuli that contain motion information but essentially no static form information. These formless dot fields differ from point-light walkers in that they can be made of thousands of dots, and they are free from the variations in dot density seen when uniform dot textures are mapped onto three-dimensional models. This technique is not the first with these properties (e.g., Sperling & Landy, 1989; Treue, Husain, & Anderson, 1991), but due to the use of hardware graphics acceleration, it is very fast. It is able to transform a modeled action sequence involving thousands of polygons into a structure-from-motion stimulus involving thousands of dots in real time. This makes it possible to present parametrically varied, actively controlled, or large numbers of individual objects without needing to pre-render an inordinate number of movies. An example of a formless dot field stimulus, created from a dancing humanoid, is shown in Movie 1.

Methods

The simplest way to describe this technique is with a physical analogy. Every surface in the scene to be rendered is invisible. Objects are composed of connected triangles, and there is an invisible wall behind all the objects in the scene. At each incremental point in time, a fixed number of darts are shot from the viewpoint onto the scene, with a distribution of trajectories that is uniform with respect to the horizontal and vertical axes (for an orthographic projection) or with respect to azimuth and elevation (for a perspective projection). For each dart that strikes a triangle, a small glowing sticker is applied to that triangle. These dots glow for a fixed, limited number of time steps, moving as the surface on which they are stuck moves. Each new set of dots, upon creation, therefore obeys a uniform distribution with respect to the observer's point of view. The back wall is also populated with a distribution of temporary glowing dots that is uniform relative to the viewer and acts as camouflage for the foreground objects.

We take advantage of OpenGL to compute which of potentially thousands of triangles is hit by each dart (also



Movie 1. A 3-D animated clip of a dancing figure has been rendered as a formless dot field. Refer to Movie 3 to see the same animated clip rendered with the figure and background in different colors.

potentially numbering in the thousands). Hardware acceleration speeds up these calculations, enabling real-time rendering of formless dot fields (other 3-D graphics framework that support hardware accelerations, such as Direct3D, could work equally well). Before we describe the algorithm, we will describe the underlying data structure.

We maintain two sets of dots, one for the dots on the metaphorical back wall and one for the dots located on foreground objects. Each dot in the background set has its position recorded as an (x, y) pair. Each dot that is on a triangle of an object has its position on that triangle stored as a set of barycentric coordinates (coordinates relative to the vertices of the triangle). Barycentric coordinates ensure that when a triangle moves, all dots on that triangle move with it, maintaining their positions on the triangle. Groups of these points are stored, one for each time step of dot lifetime. For example, if dots persist for four time steps, there will be four groups of dots in the foreground set and four in the background set. These data are created and updated as follows.

Each triangle must be assigned a fixed index; this can be as simple as ordering triangles from 1 to the total number of triangles in the scene, or by some more complex scheme that respects the organization of objects in the scene. At each time step, the scene is first rendered into an off-screen buffer (in our case using OpenGL pixel buffers, also called “pbuffers”), with each triangle’s color determined by its index. Each color therefore uniquely identifies its respective triangle; this correspondence will be used later to determine which triangles receive dots. This yields an image in which the color of each pixel indexes which triangle is visible at that pixel; the background is set to a color outside the range of the triangle indices. Lighting, texturing, and anti-aliasing are also turned off at this point, as the critical aspect of this image is the exact correspondence between particular colors and particular triangles. This image is not actually drawn on screen but is instead rendered to an off-screen buffer; see Figure 2A for an example of what this would look like, if it were actually drawn on screen. This image is saved as an array in memory and is later used to determine which triangles are struck by points. The positions of each vertex of each indexed triangle are also saved into memory.

We then create two collections of uniform random two-dimensional points, one for the new background dot group and one from which the new foreground/object dots will be drawn. This latter group is the “darts” described above. The background points simply become the newest group of the background set (note that the background points can always assume an orthographic projection with a uniform distribution, as they are “striking” empty space at an infinite distance behind the surfaces in the scene). For each potential new foreground point, the color of the corresponding pixel is accessed from the array in memory. Since colors uniquely identify triangles, this determines which triangle (if any) that point hit, taking advantage of the rendering engine’s culling

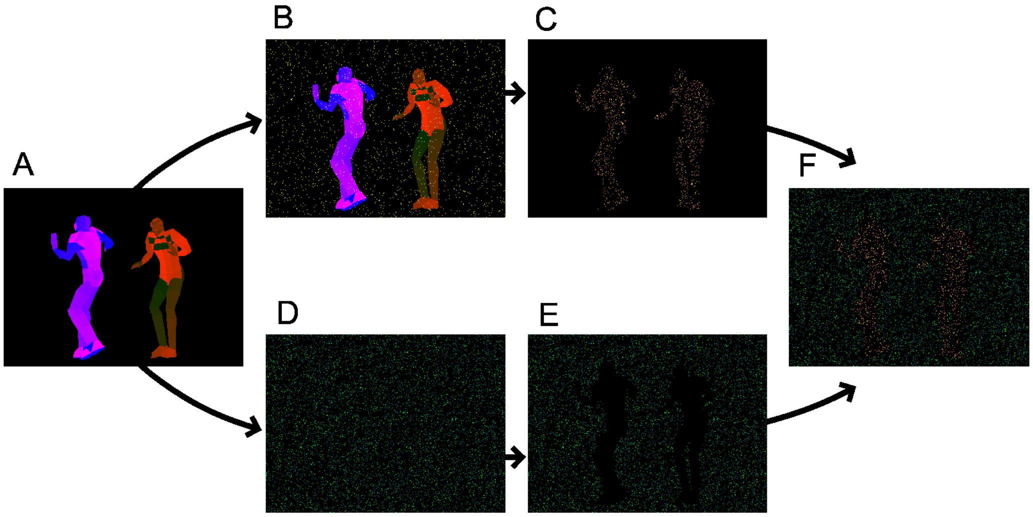


Figure 2. The process by which each formless dot field frame is updated and rendered. (A) Objects are drawn to an off-screen buffer, with each triangle a unique color. (B) A new group of uniform random points is created (yellow). The color in the off-screen buffer image at each of these points is inspected; since each triangle is uniquely associated with a color, this determines which triangle, if any, was struck by each point. Those that struck triangles will become new foreground points. Here we take advantage of the rendering engine’s culling of occluded surfaces to determine which triangle is the closest to the viewer and hence hit by the point. Points that struck triangles are associated with those triangles and are saved for later drawing. (C) The screen positions of old (pink) and new (yellow) foreground dots are calculated based on where their associated triangles are located. (D) A new group of uniform random background dots (green) is added to the remaining old dots in the background set (blue). (E) Dots in the background group are checked against the off-screen buffer image, and only those that are not occluded by foreground objects are selected to be drawn. (F) The resulting foreground dots, along with the nonoccluded background dots, are drawn.

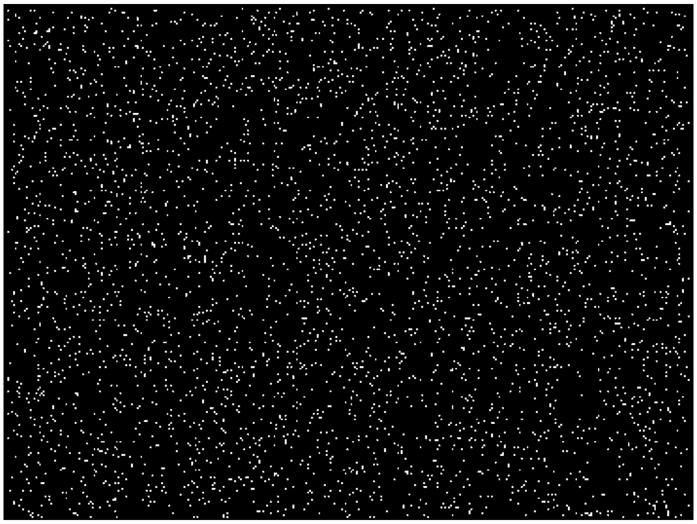
of occluded surfaces. For each point that intersects a triangle, we transform the Euclidean coordinates to barycentric coordinates within the appropriate triangle, using the saved vertex locations for that triangle. The resulting barycentric coordinates are added to the current group of the set of foreground dots. The oldest group of each set of dots is discarded, keeping the number of groups constant.

Having updated the sets of dots, we then draw them. Each foreground dot's screen position is calculated from the locations of the vertices of the enclosing triangle and the dot's barycentric coordinates. The dot is then drawn at the corresponding screen location. The background dots, for which (x, y) coordinates are already specified, are checked against the copy of the color-index image in memory. Each background dot is drawn if the corresponding pixel is the background color; otherwise, it is culled. This ensures that background dots are drawn if and only if they are not occluded by objects in the scene. After rendering is complete, the process loop begins again, creating the next sets of dots for the next time point.

The final result can be seen in Movie 2. The characters used in this and the other included movies are from Geometricks. We applied biped actions from Credo Interactive's MeGa MoCap V2 using Autodesk's 3ds Max. These were then exported as Granny 3D (RAD Game Tools, Kirkland, Washington) files, which were read and rendered by custom software.

Psychophysical test

To demonstrate that this technique produces stimuli that contain usable motion information that can also support shape judgments, as well as to probe the limits of the technique, we designed two simple psychophysical tests using the formless dot fields. Eight volunteers (including the authors) participated in these two behavioral tasks, following procedures approved by the Brown University IRB.



Movie 2. Repeated for each frame, the process outlined in Figure 2 results in a movie in which the actions of two dancing characters are visible.

Stimuli and procedures

Both tasks used the same simple set of formless dot field stimuli. The stimuli varied along six dimensions: shape (sphere or cylinder), direction of rotation (left or right), rotation speed (180°/s or 360°/s), dot lifetime (20 ms or 40 ms), strobe frequency (100 Hz full-motion or 20 Hz strobe), and number of dots (500, 4000, or 12000). All combinations of these values were included, yielding a total of 96 distinct stimuli. Trials began with a fixation point (though we did not control or monitor subjects' eye movements). The fixation point disappeared after 450 ms, and after a 500-ms blank period, one of the 96 formless dot fields appeared. The target object rotated for 2 s before stopping and becoming invisible. The formless dot field subtended approximately 10° of visual angle, with the sphere or cylinder occupying the central 6°.

In the direction task, subjects were instructed to respond according to the direction of rotation of the object (left or right). In the object task, the subjects were asked to indicate which of the two shapes was present (sphere or cylinder). We also described the artifacts that might be present in some stimuli and explained to the subjects how they could take advantage of them to solve the tasks even if they could not clearly perceive motion or form. In the direction task, the useful artifact was an increased density of dots on the side to which the shape was rotating and a corresponding decrease in density on the other side. In the object task, a straight vertical line of high or low dot density indicated the presence of a cylinder, while an arc indicated a sphere. The order in which the tasks were presented was counterbalanced across subjects.

Results

When a surface rotates away from the viewplane with a rotation speed that is fast relative to the lifetime of the dots, edge artifacts appear. A region of high dot density develops on the edge of the surface rotating away from the viewer, and a region of low dot density appears on the edge rotating towards the viewer. The stimuli used in the behavioral task were designed as a test of the worst-case scenario with regard to this issue—they are simple rigid shapes, rotating away from the viewer but otherwise stationary. Subjects were informed of the presence of these artifacts and instructed to take advantage of them whenever possible.

Because the strobe period is long enough that there are no dot correspondences between successive strobe frames, good performance in any of the strobe conditions indicates some ability to take advantage of these static cues. Subjects' performance and reaction times are summarized in Figure 3. Performance in full-motion conditions is excellent across the board, only dipping somewhat in a few of the most difficult conditions. Notably, performance drops tremendously when going from full-motion to strobe versions in almost all cases, and across a range of parameters performance in the strobe case is no better than chance. Subjects found the direction task easier than the object task; in some cases, they were able to detect nonuniformities in dot density but were unable to localize them except in a very coarse left/right manner.

Reaction times were also dramatically slower in the strobe conditions. This was even true in the one case where strobe performance approached full-motion performance (360° rotation speed, 12000 dots, 40 ms dot lifetime, object task). Except with the most extreme parameters, therefore, subjects appear unable to take easy advantage of static cues available in single frames. However, given enough of these almost-random frames, one frame will likely have some perceptible nonuniformity; more often than not, this will provide a valid cue. Alternatively, it may be that the visual system integrates information across many hundreds of milliseconds and is therefore sometimes eventually able to obtain an impression of nonuniformities in the density. This latter explanation is consistent with the results of Treue et al. (1991), who found reaction times of almost a second in two simple tasks involving a rotating cylinder described by 128 dots; they attributed these slow reaction times to the need for integration across long time periods.

Since these stimuli were constructed to maximally expose the weakness of our technique (or any similar projection technique), the performance seen here can be taken as the best a subject can do with a particular set of parameters. With a complex object moving in complex ways (in other words, with the type of stimuli for which our technique is best suited), subjects' ability to integrate over long periods of time would be vastly reduced. Moreover, it is only rotation that suffers from artifacts. Translation in the viewplane can be arbitrarily fast, up to subjects' abilities to maintain dot correspondences between frames, and has the benefit of disrupting the ability to integrate across frames at a given position.

Simulation results

We supplemented these behavioral results with an analytic test of the uniformity of the dot fields generated by a rotating sphere, rendered with 20160 dots. We did this for a range of dot lifetimes and rotation speeds. Resulting single frames can be seen in Figure 4. Only with

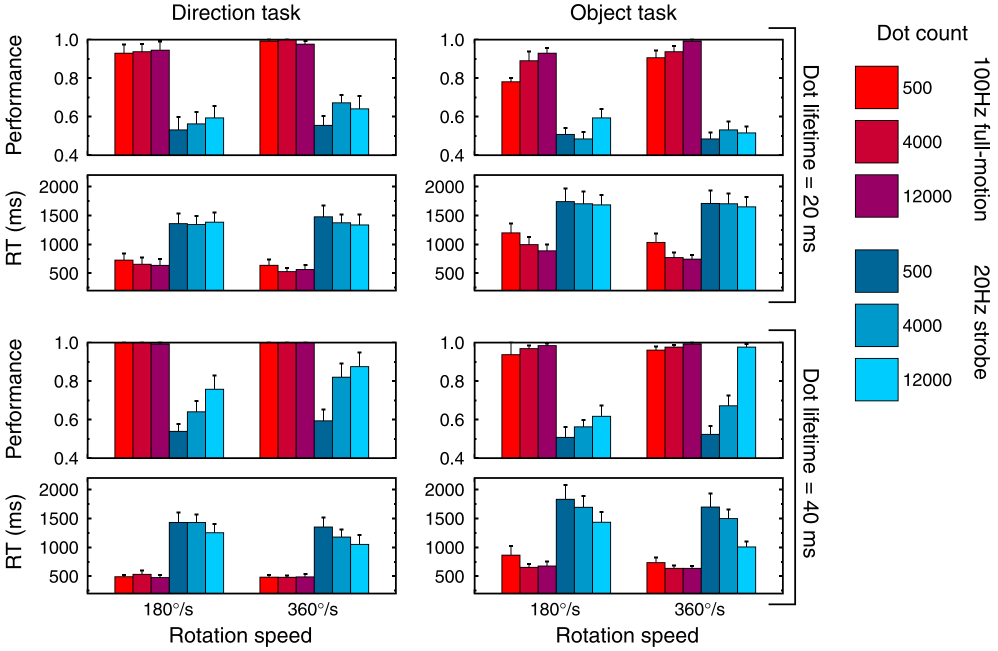


Figure 3. Results of the behavioral task for eight subjects. The left column of graphs shows results from the direction task, the right column shows results from the object task. Each box contains two sets of bars, one for the 180°/s rotations and one for the 360°/s rotations, showing the results for all dot counts and both frequencies. For each task and each dot lifetime, there are two boxes, one each for performance and reaction time. Subjects responded much sooner, and with greater accuracy, in the full-motion than in the strobe conditions across the board. For many combinations of speed, dot count, and dot lifetime, performance in strobe trials was not above chance, especially in the object task. For these sets of parameters, subjects are unable to use static cues to determine which shape is present, or even in a few cases to determine which side of the dot field contains more dots. Error bars indicate standard errors of the mean.

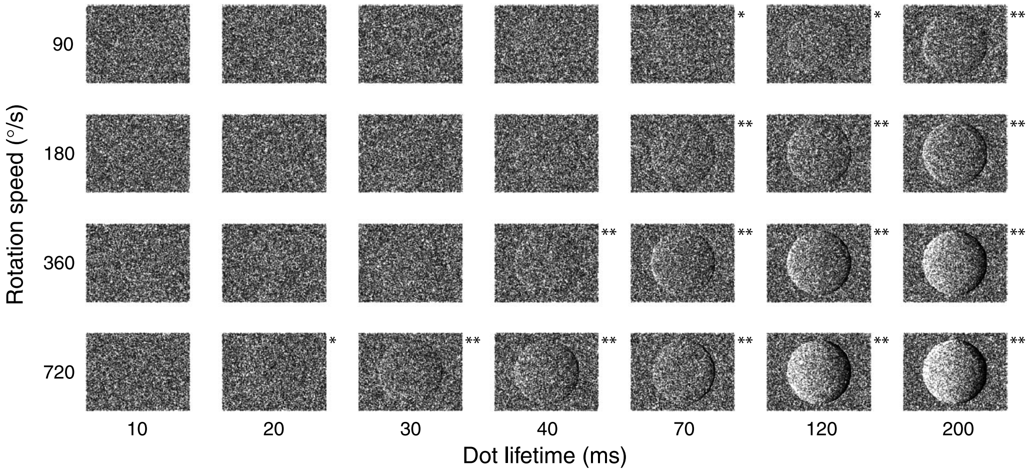


Figure 4. Single frames from movies made with our formless dot field technique. Dot lifetimes increase from left to right, and rotation speed increases from top to bottom. Even with 20160 partially transparent dots in a small area, the underlying sphere is invisible except when the dot lifetimes and/or rotation speed are very high. Rotation speeds and dot lifetimes assume a monitor with a 100-Hz refresh rate. We ran a χ^2 goodness of fit test to evaluate the null hypothesis that the frames of animation came from a uniform distribution. Probabilities are based on the average χ^2 scores from five frames of animation, each divided into 192 square regions. *Note*: *$p < 0.05$, **$p < 0.001$.

very fast rotation speed or long dot lifetimes does the underlying sphere become visible. Note that the dots used here are extremely dense; this accentuates edge artifacts and also provides an approximation of what several accumulated frames at more typical densities would look like. The dots are rendered partially transparently, so that the degree of overlap is visible. Contrast these with a uniform dot texture mapped onto a sphere (Figure 1), in which the sphere is clearly visible even at much lower dot density.

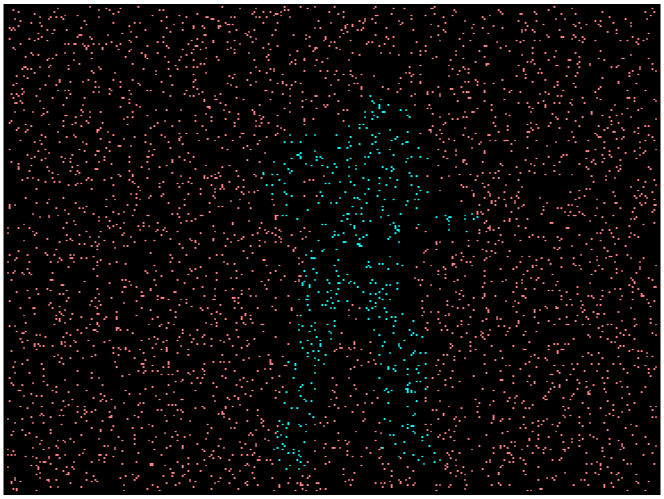
We used a χ^2 goodness of fit test to analyze how uniformly the resulting dots are distributed. For each condition's movie, we divided five separate frames each into 192 square regions. The χ^2 test gives the probability of seeing per-region dot counts as nonuniform as those observed, under the assumption that the dots are drawn from a uniform distribution. Significant deviations from uniformity, based on five frames of animation, are indicated on Figure 4. With 40 ms dot lifetimes, rotation speed must exceed 180°/s before deviations from uniformity become statistically significant, even at dot densities an order of magnitude greater than those normally used. Note that subjects in the behavioral task were able to perform better than chance in the direction task with these parameters but took well over a second to do so. Decreasing dot lifetime to 30 ms allows even 360°/s rotations without any reliable single-frame static form cues.

Discussion

This algorithm can quickly produce moving stimuli that have no apparent form in any single frame of the movie. It does so using three-dimensional models and animations, allowing not only real-time rendering but also on-the-fly modification or control of the stimulus. Despite the power of the technique, it does have some limitations.

Aside from the problems (described earlier) posed by rapidly rotating objects, there are two very minor issues with this algorithm. First, foreground dots persist even if they become occluded by other triangles. This does not occur with background dots because with background dots it is sufficient to check whether anything in the off-screen buffer occludes the dot. With foreground dots, however, we would have to check whether the location of the dot in question is occupied by a different triangle in the off-screen buffer. This is, however, not well defined—the "triangles" are actually collections of square pixels, and the edges between triangles are therefore not in general straight lines. Corners of pixels from one triangle may protrude into areas that are within the theoretical boundaries of another triangle. Any dots in these regions would not be drawn if we performed this kind of occlusion testing, resulting in a noticeable decrease in dot density when triangles are small. Overlap of dots when one triangle moves to occlude another only becomes noticeable when triangles move extremely rapidly relative to the dot lifetime. For example, all three movies presented here include many instances of self-occlusion, without any obvious local increases in dot density as a result.

The second issue is that the number of dots actually drawn is not guaranteed to be exactly the number requested since the foreground and the background sets are independent. We made this choice because the obvious alternative, using the same points to generate both sets, results in one of two undesirable effects at the trailing edges of objects. If the background region is left empty, voids indicate where objects have recently vacated. If we instead place dots at one location in both



Movie 3. The foreground and background of this formless dot field are rendered in different colors, making the dancer from Movie 1 easily separable from the background.

foreground and background, dots that overlay formerly occluded regions of background will split into two dots, one remaining motionless while the other moves to a nearby point. This could interfere with the translational apparent motion of these dots.

The distinction between foreground and background also very easily allows for the different sets of dots to be drawn in different colors. This explicit color-based segmentation into figure and ground makes perception of global form much easier (Movie 3) and allows for study of motion perception with and without global form in otherwise identical stimuli. It would also be straightforward to add more colors, thereby segmenting different objects both from each other and from the background.

Perception of motion and perception of form are central to our visual experience. The relative roles that each plays in high level vision are still a matter of considerable debate; how we perceive complex articulated motions, for example, remains an open question. Do our brains analyze sequences of snapshots, unconsciously interpolating between them? Do they extract form information from complex motion vector fields, and then use the resulting sequences of poses? Or do they operate directly on the complex motion vector fields themselves? Careful experiments using stimuli like those created by this algorithm have the potential to answer these and related questions.

Conclusions

Here we have presented a structure-from-motion technique that can eliminate any useful contribution of form present in individual frames. That the resulting movies are as vivid as they are reaffirms that it is possible to extract considerable information about an object's form from correspondences in successive frames of otherwise unstructured dots. In addition to the lack of form information, the real-time nature of this technique offers new experimental opportunities. For example, staircase and matching procedures, sometimes impractical when pre-rendered movies are required, are simple to implement when the underlying models can be parametrically varied. Even more complex interaction between subject and stimulus can easily be achieved; subjects could guide an animated figure though a complex environment, with both figure and environment visible only by virtue of their apparent motion. By segmenting foreground and background with color, it is possible to reintroduce explicit (global) form; by preserving or eliminating motion, all possible combinations of form and motion information can be generated. We anticipate that these formless dot field stimuli will be a useful tool in the study of motion perception and its interaction with form perception.

Acknowledgments

This research was supported by National Institutes of Health Grant EY-014681, Grant Number NCRR C06-16549-01A1, NSF SBE-0542013, and the James S. McDonnell Foundation.

Commercial relationships: none.

Corresponding author: David L. Sheinberg.
Email: David_Sheinberg@brown.edu.
Address: Brown University, Box G-LN, Providence, RI 02912, USA.

References

Beintema, J. A., & Lappe, M. (2002). Perception of biological motion without local image motion. *Proceedings of the National Academy of Sciences of the United States of America, 99,* 5661–5663. [PubMed] [Article]

Caplovitz, G., & Tse, P. (2007). Rotating dotted ellipses: Motion perception driven by grouped figural rather than local dot motion signals. *Vision Research, 47,* 1979–1991. [PubMed]

Casile, A., & Giese, M. A. (2005). Critical features for the recognition of biological motion. *Journal of Vision, 5*(4):6, 348–360, http://journalofvision.org/5/4/6/, doi:10.1167/5.4.6. [PubMed] [Article]

Cowey, A., & Vaina, L. M. (2000). Blindness to form from motion despite intact static form perception and motion detection. *Neuropsychologia, 38,* 566–578. [PubMed]

Domini, F., & Caudek, C. (1999). Perceiving surface slant from deformation of optic flow. *Journal of Experimental*

Psychology: Human Perception and Performance, 25, 426–444. [PubMed]

Doner, J., Lappin, J. S., & Perfetto, G. (1984). Detection of three-dimensional structure in moving optical patterns. *Journal of Experimental Psychology: Human Perception and Performance, 10,* 1–11. [PubMed]

Giese, M. A., & Poggio, T. (2003). Neural mechanisms for the recognition of biological movements. *Nature Reviews, Neuroscience, 4,* 179–192. [PubMed]

Hoffman, D. D., & Flinchbaugh, B. E. (1982). The interpretation of biological motion. *Biological Cybernetics, 42,* 195–204. [PubMed]

Johansson, G. (1973). Visual perception of biological motion and a model for its analysis. *Perception & Psychophysics, 14,* 201–211.

Sperling, G., & Landy, M. S. (1989). Kinetic depth effect and identification of shape. *Journal of Experimental Psychology: Human Perception and Performance, 15,* 826–840. [PubMed]

Treue, S., Husain, M., & Andersen, R. A. (1991). Human perception of structure from motion. *Vision Research, 31,* 59–75. [PubMed]

Tse, P. U., & Logothetis, N. K. (2002). The duration of 3-D form analysis in transformational apparent motion. *Perception & Psychophysics, 64,* 244–265. [PubMed]

Vaina, L. M., Lemay, M., Bienfang, D. C., Choi, A. Y., & Nakayama, K. (1990). Intact "biological motion" and "structure from motion" perception in a patient with impaired motion mechanisms: A case study. *Visual Neuroscience, 5,* 353–369. [PubMed]

van Damme, W. J., & van de Grind, W. A. (1996). Non-visual information in structure-from-motion. *Vision Research, 36,* 3119–3127. [PubMed]

Wallach, H., & O'Connell, D. N. (1953). The kinetic depth effect. *Journal of Experimental Psychology, 45,* 205–217. [PubMed]