Analyzing Intrinsic Motion Textures Created from Naturalistic Video Captures

Angus Graeme Forbes[1], Christopher Jette[1] and Andrew Predoehl[2]
[1]*School of Information: Science, Technology, and Arts, University of Arizona, Tucson, U.S.A.*
[2]*Department of Computer Science, University of Arizona, Tucson, U.S.A.*

Keywords: Intrinsic Motion Textures, Psychophysics, Perception, Metrics for Motion, Dynamic Textures.

Abstract: This paper presents an initial exploration of the plausibility of incorporating subtle motions as a useful modality for encoding (or augmenting the encoding of) data for information visualization tasks. Psychophysics research indicates that the human visual system is highly responsive to identifying and differentiating even the subtlest motions intrinsic to an object. We examine aspects of this *intrinsic* motion, whereby an object stays in one place while a texture applied to that object changes in subtle but perceptible ways. We hypothesize that the use of subtle intrinsic motions (as opposed to more obvious *extrinsic* motion) will avoid the clutter and visual fatigue that often discourages visualization designers from incorporating motion. Using transformed video captures of naturalistic motions gathered from the world, we conduct a preliminary user study that attempts ascertains the *minimum* amount of motion that is *easily perceptible* to a viewer. We introduce metrics which allow us to categorize these motions in terms of *flicker* (local amplitude and frequency), *flutter* (global amplitude and frequency), and average maximum *contrast* between a pixel and its immediate neighbors. Using these metrics (and a few others), we identify plausible ranges of motion that might be appropriate for visualization tasks, either on their own or in conjunction with other modalities (such as color or shape), without increasing visual fatigue. Based on an analysis of these initial preliminary results, we propose that the use of what we term "intrinsic motion textures" may be a promising modality appropriate for a range of visualization tasks.

1 INTRODUCTION

Rustling leaves, flickering flames, sunlight sparkling on water – every day we are continually confronted with naturalistic motion as we navigate the world. Information visualization, as a field, examines how meaning is effectively conveyed through visually-encoded data. However, dynamic visualizations that encode data using motion have not been as widely explored, as motion is generally considered to be too distracting a modality for representing information effectively. The human visual system is exceptionally adept at identifying differences in texture (Emrith et al., 2010), and observing both the movement of objects or the movement within an object (Chalupa et al., 2004). While far from being fully understood, a growing body of research indicates the presence of multiple neurological mechanisms for processing *extrinsic* motion (the movement of an object) and *intrinsic* motion (local movement within a single object) (Lu and Sperling, 2001; Nishida et al., 1997). Extrinsic and intrinsic motion is also termed "first-order" or "second-order," respectively: first-order motion referring to the perception of a change in luminance across the visual field; second-order motion referring instead to the perception of changes in texture or contrast. While both types of motion can indicate a global movement of a single entity, in this paper we examine the use of second-order motion to indicate motions within a stationary object. We introduce a method for transforming real-world motion into abstract motion textures with subtle motions, which we are calling *intrinsic motion textures*. These transformed video captures of real-world naturalistic motion allow us to experiment with non-distracting motion without including their representational aspects. Through the transformation of, say, the movement of water in a stream, we are able to capture the motion without directly reminding users that they are looking at water. We also introduce an easy-to-calculate set of metrics to characterize these intrinsic motions. While the vast range of possible motions makes it rather daunting to attempt to encompass all types of movement via a single set of metrics, ours capture the main features of

Graeme Forbes A., Jette C. and Predoehl A..
Analyzing Intrinsic Motion Textures Created from Naturalistic Video Captures.
DOI: 10.5220/0004660401070113
In *Proceedings of the 5th International Conference on Information Visualization Theory and Applications* (IVAPP-2014), pages 107-113
ISBN: 978-989-758-005-5

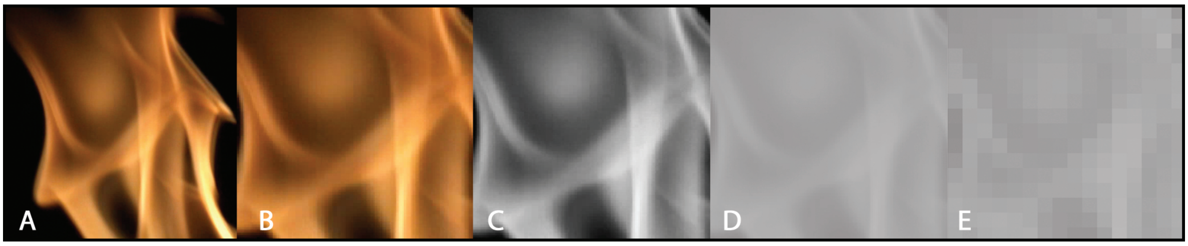

Figure 1: Stages of the video capture of naturalistic motion being transformed into an intrinsic motion texture. Here we show a single frame (A) as it is (B) cropped, (C) desaturated, (D) contrast-compressed with a low pixel range, and (E) pixelated. A further step uses temporal smoothing to mitigate extra flickering introduced from the pixelation.

intrinsic motion. We conducted a user study asking participants to evaluate a set of intrinsic motion textures. We were primarily interested in gathering insight into the following question – *What is the least amount of motion needed in order to easily identify differences between highly similar motions?* We discovered that even intrinsic motion textures with a low contrast range are easily distinguishable, as long as certain amounts of *flicker*, *flutter*, and average maximum *contrast* (defined below) are present.

In the influential text, "Semiology of Graphics," Jacques Bertin introduces a typology of retinal variables applicable to the communication of information. Although explicitly discussing static print media, Bertin hints at the ability of texture, unlike other retinal variables, to produce a vibratory effect that is the "collusion" of physiological and psychological effects. Although he does not explore this issue of vibratory effects further, he encourages designers "to make the most of this variation, to obtain the resonance without provoking an uncomfortable sensation, to flirt with ambiguity without succumbing to it." Bertin speaks of texture in terms of the varying thicknesses of lines, points, or shapes to indicate an overview similarity or differentiation between different qualities, or an ordered-ness within the same quality. However, he does not believe that the use of texture is refined enough to allow viewers to perceive proportionality and therefore is not effective at allowing users to perform quantitative tasks using texture alone (Bertin, 2010).

Nonetheless, investigations into the use of static textures in visualization contexts have found that they can be effective for representing quantitative data in certain tasks. A seminal paper (Ware and Knight, 1995) uses Gabor patches to parametrically investigate aspects of texture that may be useful in information display, such as orientation, size, and contrast. Applied research introduced in (Kujala and Lukka, 2003) explores the use of a "parameter hierarchy" in the procedural creation of perceptually distinct textures for the effective display of information. More recently, (Byelas and Telea, 2009) uses overlapping textures to indicate multivariate data in software diagrams, explicitly attempting to indicate an increased number of variables within the diagram. (House et al., 2006) describes a technique for the perceptual optimization of complex visualizations involving layered textures. And (Interrante and Kim, 2001) and (Kim et al., 2004) explore the efficacy of using various orientations and types of textures to facilitate shape perception. These examples are mostly explicitly concerned with the potential expanded range of variables that can be expressed, while at the same time aware that the cost of this expanded range might be perceptual fatigue or cognitive overload, or worse, an inability to clearly distinguish differences. (Interrante, 2000) examines the use of overlapping naturalistic textures to indicate multivariate data while mitigating against the "extraneous stress" that might occur with synthesized textures. In addition to being a potentially effective representation of a wide range of quantitative information, the use of certain aspects of naturalistic textures, such as variability, might be used to indicate extra information, such as uncertainty. Interestingly, recent perceptual experiments, such as (Emrith et al., 2010), confirm that humans perceive even minute alterations in texture, but note that it is in fact somewhat easier to discern differences between synthetic textures than natural textures.

Motion is often used to signal transitions between views and contexts, to signal interruptions, and to indicate temporal aspects of data. However, it is less frequently used as an encoding mechanism for quantitative or qualitative information. (Forbes et al., 2010) presents a data visualization framework than enables animation to be mapped to dynamic streams of data, and (Bostock et al., 2011) describes a framework that includes "transition" operators for animating data points. The use of motion in visualization elicits concern about visual clutter and perceptual fatigue, even while potentially providing an expanded toolset for representing information. At its most extreme, the injudicious use of motion in information might cause significant visual stress (Ware, 2004). One group of security researchers (Conti et al., 2005) even describes the potential for malicious hackers to takeover an information visualization system and alter its visual output to induce epileptic seizures.

Results from (Bartram and Ware, 2002) show that small, brief, and graphically simple extrinsic motions are perceptually efficient ways to distinguish objects in a crowded display. In particular, they note that a synchronization of elements is required in order for them to be effectively recognized as similar. That is, the timing of the motion is as important as the motion itself. Research into extrinsic motion cues, or "moticons" (Bartram et al., 2003), finds that motion coding is independent from color and shape coding and that more subtle motions are less distracting to users yet easily perceived. A series of experiments that analyzed extrinsic aspects of motion– velocity, direction, and on-off blinking– finds that these properties are all effective at encoding multiple data values in a prototype astrophysics simulation, provided they meet certain basic thresholds of perceptibility (Huber and Healey, 2005). A technique termed "motion highlighting" explores the potential applicability of motion to node-link diagrams (Ware and Bobrow, 2004). Results of motion highlighting experiments indicate that the translating or scaling of node is more useful for supporting rapid interactive queries on node-link diagrams than static highlighting methods.

Dynamic textures are sequences of images that exhibit some form of temporal coherence, or more specifically, they are individual images that are "realizations of the output of a dynamical system driven by an independent and identically distributed process" (Doretto et al., 2003). Dynamic textures have been effectively used in scientific visualizations and have been more extensively investigated in computer graphics and computer vision research. For instance, (Van Wijk, 2002) uses an iterative series of texture distortions to represent fluid flows, and (Forbes and Odai, 2012; Forbes et al., 2013) applies this technique to creative media arts projects. Work by (Lum et al., 2003) explores adding moving particles to a surface texture of a static object in which the particles are placed along the principal curvature direction to better indicate the object's shape and spatial relationships. Within a more general computer graphics context, dynamic textures are used in a variety of applications. For instance, dynamic textures have been used as a computationally efficient way to add realism to a scene. (Chuang et al., 2005) presents an interactive system that procedurally generates dynamic textures from selected components of a single image that can then be added to a scene. Similarly, a variety of techniques have been introduced to automatically create "temporal textures" from a single image in order to mimic natural phenomenon such as clouds, water, and fire (Lai and Wu, 2007; Ruiters et al., 2010; Okabe et al., 2011). In addition to having the potential to be used as an effective modality for representing quantitative information, recent research has explored the use of dynamic textures as a medium for providing semantically contextualized information. (Lockyer et al., 2011) explores the "expressive scope" of ambient motion textures for "emphasis and more subtle ambient visualization." In particular, this research focused on the effective communication of particular emotions through the use of intrinsic motion cues within a dynamic texture.

For the most part, research on motion in information visualization is concerned with extrinsic motion, or at least does not differentiate between extrinsic and intrinsic motion. For instance, (Ware and Bobrow, 2004), also cited above, discusses a motion highlighting technique whereby the animation of a stationary link generate a "crawling" motion. Although it is not presented specifically as a dynamic texture, it is clear that this "crawling" motion is of a different nature than the translation patterns used to highlight nodes. A recent evaluation found that animated representations were more effective than almost all static representations of link representations (Holten et al., 2011). It seems reasonable that other visualization systems could utilize a conflation of textures and motion; rather than attempting to procedurally generate dynamic textures, we could gather them directly. This would have the immediate advantage that they were, at least in some degree, inherently non-distracting for the simple reason that they occur continually in the real-world. An earlier (unpublished) study by one of the authors found that the use of moving sinusoidal gratings introduced visual fatigue precisely because of the qualities that made it unrealistic: its fixed rotation, its predictable frequency and amplitude, and its repetitive sequence of pixel values. instead, Dynamic textures using fluctuating, intrinsic, real-world motion are cohesive without being repetitive; differentiable without being distracting.

In order to expedite the creation of intrinsic motion textures in order to analyze their potential effectiveness in visualization systems, we gathered real-world video of natural phenomena containing intrinsic motion: fire, water, clouds, etc. Although we believe that these textures have a representational component that might be useful in some visualization circumstances, for this study we isolated the phenomena using these steps: (A) record the natural phenomenon; (B) crop the resulting video; (C) desaturate the video;

(D) constrain the pixel range of the video; (E) pixelate the video; (F) apply temporal smoothing to the video. The video is thus transformed into a low-contrast, desaturated, pixelated, and mostly unrecognizable version of itself that nonetheless retains important qualities of the original natural motion. Figure 1 shows a frame from a naturalistic video as it is processed through this pipeline. These intrinsic motion textures can then be defined metrically, in terms of particular features, and included in studies where we can associate these features with empirical observations, such as discernibility and differentiability.

2 METRIC DEFINITIONS

Much research has been done to develop ways to effectively and efficiently characterize motion. A commonly used method, optical flow, assumes that there is a unique velocity vector at each pixel. Other methods relax that assumption. For instance, (Langer and Mann, 2003) introduces "optical snow," which is able to characterize motions that have prevalent discontinuities between frames. The intrinsic motions that we have gathered likewise include large amounts of flickering that are not captured with optical flow type analyses. Since we are, for now, interested primarily in the single task of determining the ease of discrimination betweens motions, we constructed a way to create simpler metrics that define a video via a set of eight features that sufficiently characterize intrinsic motions. They are grouped into the following categories: *flicker*, *flutter*, and *variability*.

We now introduce notation that defines our metrics precisely. Let $I_{q,t}$ denote the integer intensity value of a pixel in the video, at spatial position q and at time t. Assuming the video pixels lie in a rectangular grid of width W, height H, and that the video has duration of T discrete video frames, $q \in Q = \{1,\ldots,W\} \times \{1,\ldots,H\}$ and $t \in \{1,\ldots,T\}$. The first characteristic of interest we define is the *contrast range* K of a video:

Definition 1 (Contrast range)**.**

$$K = \max_{\substack{q\in Q\\ 1\le t\le T}} \left\{I_{q,t}\right\} - \min_{\substack{q\in Q\\ 1\le t\le T}} \left\{I_{q,t}\right\}.$$

In the present study, instead of using K as a video feature for comparison, we partitioned our test videos into collections with of similar contrast range, because two videos with widely differing contrasts are very obviously distinct. Specifically, we gathered videos with contrast ranges of 20 or less, 21 to 40, 41 to 60, and 61 to 80, into groups called G_{20}, G_{40}, G_{60}, and G_{80}, respectively.

The variability metrics reflect the spatial variation in a single video frame. Let $Q' \subset Q$ be the set of pixel positions in the interior of the grid. Each pixel position $q \in Q'$ therefore has eight spatially adjacent neighbors, the set of which we denote $N(q)$. The *roughness* R is defined as the average intensity difference of the highest-contrast neighbors:

Definition 2 (Roughness)**.**

$$R = \frac{1}{WH}\cdot\frac{1}{T}\sum_{q\in Q'}\sum_{t=1}^{T}\max_{s\in N(q)}\left|I_{q,t} - I_{s,t}\right|.$$

Edginess E_θ is the average number of large-contrast juxtapositions, per pixel, per frame. The contrast is regarded as large if the intensity difference is at least θ:

Definition 3 (Edginess)**.**

$$E_\theta = \frac{1}{WH}\cdot\frac{1}{T}\sum_{q\in Q'}\sum_{t=1}^{T} 1_{\{\tau\,:\,\theta\le\max_{s\in N(q)}|I_{q,\tau}-I_{s,\tau}|\}}(t).$$

In the current study, we set the threshold value θ equal to one-eighth the maximum contrast range of the relevant collection (G_{20}, G_{40}, G_{60}, or G_{80}). For example, when comparing two videos from group G_{40}, we used the $E_{40/8} = E_5$ edginess metric.

Our flicker metrics depend on the local maxima and minima (peaks and valleys) of pixel intensity in the time domain. To specify them formally, we introduce definitions of *peak* and *valley* as follows.

Definition 4 (Peak intensity value)**.** $I_{q,t}$ is a *peak intensity value of width* j, provided

- $j > 0$,
- $1 < t \le T - j$,
- $I_{q,t-1} < I_{q,t} = I_{q,t+1} = \cdots = I_{q,t+j-1}$, and
- $I_{q,t+j-1} > I_{q,t+j}$.

Though cumbersome, this definition is consistent with an intuitive notion of local maximum. We also define *valley intensity value* analogously for local minima. Let n_q denote the number of peaks of width 1 or more at position q. Let $p_{1,q}, p_{2,q}, \ldots, p_{n_q,q}$ be the peak intensity values at position q, such that $p_{i,q}$ is the intensity value of the ith peak, in chronological order. Similarly, let $v_{1,q}, v_{2,q}, \ldots, v_{n_q,q}$ denote the valley intensity values (assuming they are equally numerous as the peaks). Now we can precisely state our metrics for flicker. First, the average number of peaks per pixel, per frame, is called *local frequency*, F_L:

Definition 5 (Local frequency)**.**

$$F_L = \frac{1}{WH}\cdot\frac{1}{T}\sum_{q\in Q} n_q.$$

We define the average peak-to-valley difference, per pixel, as *local amplitude*, A_L:

Definition 6 (Local amplitude)**.**

$$A_L = \frac{1}{WH} \sum_{q \in Q} \frac{1}{n_q} \sum_{i=1}^{n_q} (p_{i,q} - v_{i,q}),$$

if $n_q > 0$, otherwise $A_L = 0$.

We define *local choppiness*, $C_{L,\theta}$, as the average number of large intensity jumps per pixel, per frame. A jump at position q is large if equals or exceeds threshold value θ.

Definition 7 (Local choppiness)**.**

$$C_{L,\theta} = \frac{1}{WH} \cdot \frac{1}{T} \sum_{q \in Q} \sum_{t=2}^{T} 1_{\{\tau : |I_{q,\tau} - I_{q,\tau-1}| \geq \theta\}}(t)$$

Similar to our use of Def. 3, in our study we set the threshold value here to one-eighth of the contrast range of whichever videos we are comparing. So when comparing videos from group G_{40}, we use local choppiness metric $C_{L,5}$.

Our flutter metrics are similar to the flicker metrics, but they depend on the average intensity of the entire video frame at a given moment. Let J_t denote the average pixel intensity of the frame at time t, i.e., $J_t = \frac{1}{WH} \sum_{q \in Q} I_{q,t}$. This sequence also has peaks and valleys, which we assume are m in number. At the risk of confusion, we will denote its peak values and valley values (regardless of width) as $p_1, p_2, \ldots, p_m$ and $v_1, v_2, \ldots, v_m$ respectively, in chronological order. (Note that these values have only one subscript.) The average number of these peaks, per frame, is called *global frequency*, F_G:

Definition 8 (Global frequency)**.**

$$F_G = m/T.$$

The average of these peak-to-valley differences is the *global amplitude*, A_G:

Definition 9 (Global amplitude)**.**

$$A_G = \frac{1}{m} \sum_{i=1}^{m} (p_i - v_i),$$

if $m > 0$, otherwise $A_G = 0$.

We define *global choppiness* $C_{G,\theta}$ as the average number of large increases in average intensity, with threshold θ:

Definition 10 (Global choppiness)**.**

$$C_{G,\theta} = \frac{1}{T} \sum_{t=2}^{T} 1_{\{\tau : |J_\tau - J_{\tau-1}| \geq \theta\}}(t).$$

Again, in our study we set the this metric's threshold just as those of Defs. 3 and 7. So, when comparing videos from group G_{40}, we compute metric $C_{G,5}$.

These metrics are easy to calculate and capture both the global and the local, pixel level aspects of the videos. They may not however capture larger-scale movement within the motion textures. However, since our express aim is to use motion textures that do not contain large-scale motion, we believe that these metrics are appropriate as a first attempt to characterize intrinsic motions.

3 USER STUDY

The main goal of our preliminary user study was to determine the minimum amount of movement required in order for a participant to quickly differentiate between similar motions. Since motion can be highly distracting and since humans are exceptionally good at noticing differences in motion, by finding lower bounds on various parameters that make motion distinguishable we can identify the minimum values of easily-discernible features. Future work will use a more rigorously defined empirical study using techniques to measure just-noticeable difference, as well as explore user response to visualization tasks incorporating motion textures. For this preliminary study we wanted to obtain an initial sense of what attributes were most easily noticeable at low-contrast ranges, and which of these attributes were thought to be the least distracting.

To find this minimum feature set, we created a study that presented the participant with a pair of videos. The user was then asked to indicate whether he or she agreed or disagreed with a series of statements about the videos. We gathered 32 unique videos of naturalistic motion and processed them as described in section 3.1. We created 4 "bins" and made versions of each of these videos with different levels of contrast. Bin 1 contained videos with a contrast range of +/- 10; Bin 2, +/- 20; Bin 3, +/- 30; Bin 4, +/- 40. For Bin 1, it was very difficult to tell most of the videos apart, especially when looking at a single (unmoving) frame from the video. In other words, the contrast was so low that without movement it would be almost impossible to tell them apart. For Bin 2, it seemed that about half of the time it was easy to tell the videos apart and the other half of the time it was difficult. For Bin 3, it became much easier to tell any of the videos apart from any of the others. And finally, for Bin 4 it was easy to tell all of the videos apart. However, we thought that if the movements became more chaotic (higher absolute flicker amplitude and frequency) then in those cases the videos in Bin 3 and Bin 4 would be hard to tell apart. We did not test any of the videos against a video with a different range of pixel values as it is easy to discern the differences in videos when one had a higher maximum and lower minimum pixel value than the other. Each of the videos was analyzed with custom software that output the features described by our metrics system. We further calculated the absolute difference between the feature vectors of each video.

We included a series of four Likert items per test designed to elicit the participant's opinion about the discernibility of flicker and frequency. We ran various "batches" of our test over the course of a week and a half on Amazon Mechanical Turk. We received a total of 144 completed studies. For most of these batches, we randomly chose one of the 4 bins for each test. The majority of our "workers," 107, indicated that they were from India; 24 were from the United States; the rest came from the United Kingdom, Mexico, Sri Lanka, Canada, Pakistan, and Nigeria. 2 participants chose "Other" as their nationality. There were an equal number of male and female participants (72 each). The minimum and maximum age was 19 and 63, respectively, with a median age of 31. Following the suggestions in (Heer and Bostock, 2010), which describes some of the advantages and disadvantages of conducting studies via Mechanical Turk, we made a substantial effort to encourage reliable participation and mitigate inaccurate or random answers, ultimately obtaining 476 samples from the 144 participants.

All features, except for the frequency of the flutter (the global frequency of a direction change in average pixel value for a frame) were positively correlated with easy differentiability. We built a statistical model to characterize the relationship between the video motion metrics and the participants' responses. We focused on contrast ranges 20 and 40 and modeled the data as a two-category classification problem: in each video comparison, the videos are either difficult to distinguish (category $\mathcal{C}_D$) or not (category $\mathcal{C}_E$), generated by the user's Likert responses. For our binary classifier, any value greater than 2 was given a placed in category $\mathcal{C}_E$, and any value less than or equal to 2 was placed in category $\mathcal{C}_D$. Ideally, the model would effectively predict the category for a video comparison, based only on our video features. We modeled these two categories by assuming a multivariate Gaussian distribution of the feature vectors (which are the absolute differences of each of the eight metrics for the pair of videos being compared). In other words, we computed the maximum-likelihood mean vector $\mu_{C,r}$ and covariance matrix $\Sigma_{C,r}$ of all feature vectors for category $C \in \{\mathcal{C}_D, \mathcal{C}_E\}$, and contrast range $r \in \{20, 40\}$. For the purpose of classification, we also make use of the empirical frequency of $\mathcal{C}_D$ and $\mathcal{C}_E$ classes, denoted $p(\mathcal{C}_D)$ and $p(\mathcal{C}_E)$. Given a new data vector x for contrast range r, we would classify it in category $\mathcal{C}_D$ provided it satisfies $p(\mathcal{C}_D|x) > p(\mathcal{C}_E|x)$, where by Bayes' theorem, for $C \in \{\mathcal{C}_D, \mathcal{C}_E\}$,

$$p(C|x) = \frac{\mathcal{N}\left(x; \mu_{C,r}, \Sigma_{C,r}\right) p(C)}{\sum_{B \in \{\mathcal{C}_D, \mathcal{C}_E\}} \mathcal{N}\left(x; \mu_{B,r}, \Sigma_{B,r}\right) p(B)}.$$

In the above, $\mathcal{N}(x; \mu, \Sigma)$ denotes the probability density function for the multivariate Gaussian with mean μ and covariance Σ. One advantage of a Gaussian characterization is that we can easily marginalize any subset of features. Thus we can see the average interaction between any two features and can list the thresholds for the classifier with all other features marginalized (Table 1). In particular, even small differences in flickering (especially in the frequency and choppiness) at the individual pixel level were the main predictors of whether or not a video pair was likely to be easily differentiable.

Table 1: Threshold values between features.

l_chop	0.0011901
l_amp	0.12419
l_freq	0.00081854
g_chop	0.048523
g_amp	0.037790
g_freq	0.0056176
rough	0.57185
edge	0.0033416

4 CONCLUSIONS

This paper presents an initial foray into exploring the potential usefulness of intrinsic motion textures. We provide a method for generating cohesive, non-repetitive, intrinsic motions textures from real-world video captures; a method for characterizing the features of intrinsic motions; a preliminary user study that indicates minimal differences necessary for differentiation between motions; and an analysis of this study that identifies thresholds on these features. This initial exploration of motion textures created from video captures of naturalistic movement seems to indicate that this may be a promising area for future investigations. Future work will involve the design and analysis of more rigorous empirical studies to determine the validity of our claims regarding the noticeability and distraction of these types of textures.

REFERENCES

Bartram, L. and Ware, C. (2002). Filtering and brushing with motion. *Information Visualization*, 1(1):66–79.

Bartram, L., Ware, C., and Calvert, T. (2003). Moticons: detection, distraction and task. *International Journal of Human-Computer Studies*, 58(5):515–545.

Bertin, J. (2010). *Semiology of graphics: diagrams, networks, maps*, pages 79–81. Esri Press.

Bostock, M., Ogievetsky, V., and Heer, J. (2011). D^3: Data-driven documents. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 17(12):2301–2309.

Byelas, H. and Telea, A. (2009). Visualizing multivariate attributes on software diagrams. In *Software Maintenance and Reengineering, 2009. CSMR '09. 13th European Conference on*, pages 335 –338.

Chalupa, L., Werner, J., and of Technology, M. I. (2004). *The visual neurosciences*. MIT Press.

Chuang, Y., Goldman, D., Zheng, K., Curless, B., Salesin, D., and Szeliski, R. (2005). Animating pictures with stochastic motion textures. *ACM Transactions on Graphics (TOG)*, 24(3):853–860.

Conti, G., Ahamad, M., and Stasko, J. (2005). Attacking information visualization system usability overloading and deceiving the human. In *Proceedings of the 2005 symposium on Usable privacy and security*, pages 89–100. ACM.

Doretto, G., Chiuso, A., Wu, Y. N., and Soatto, S. (2003). Dynamic textures. *International Journal of Computer Vision*, 51:91–109. 10.1023/A:1021669406132.

Emrith, K., Chantler, M., Green, P., Maloney, L., and Clarke, A. (2010). Measuring perceived differences in surface texture due to changes in higher order statistics. *JOSA A*, 27(5):1232–1244.

Forbes, A. G., Höllerer, T., and Legrady, G. (2010). Behaviorism: A framework for dynamic data visualization. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 16(6):1164–1171.

Forbes, A. G., Höllerer, T., and Legrady, G. (2013). Generative fluid profiles for interactive media arts projects. In *Proceedings of the International Symposium on Computational Aesthetics in Graphics, Visualization, and Imaging (CAe)*, pages 123–129, Anaheim, California.

Forbes, A. G. and Odai, K. (2012). Iterative synaesthetic composing with multimedia signals. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 573–578, Ljubjiana, Slovenia.

Heer, J. and Bostock, M. (2010). Crowdsourcing graphical perception: using mechanical turk to assess visualization design. In *ACM Human Factors in Computing Systems (CHI)*, pages 203–212. ACM.

Holten, D., Isenberg, P., van Wijk, J., and Fekete, J. (2011). An extended evaluation of the readability of tapered, animated, and textured directed-edge representations in node-link graphs. In *Pacific Visualization Symposium (PacificVis), 2011 IEEE*, pages 195 –202.

House, D., Bair, A., and Ware, C. (2006). An approach to the perceptual optimization of complex visualizations. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 12(4):509–521.

Huber, D. and Healey, C. (2005). Visualizing data with motion. In *Proceedings of IEEE Visualization (VIS)*, pages 527–534.

Interrante, V. (2000). Harnessing natural textures for multivariate visualization. *Computer Graphics and Applications, IEEE*, 20(6):6 –11.

Interrante, V. and Kim, S. (2001). Investigating the effect of texture orientation on the perception of 3d shape. In *Human Vision and Electronic Imaging VI*, volume 4299, pages 330–339.

Kim, S., Hagh-Shenas, H., and Interrante, V. (2004). Conveying shape with texture: Experimental investigations of texture's effects on shape categorization judgments. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 10(4):471–483.

Kujala, J. and Lukka, T. (2003). Rendering recognizably unique textures. In *Information Visualization, 2003. IV 2003. Proceedings. Seventh International Conference on*, pages 396–405. IEEE.

Lai, C. and Wu, J. (2007). Temporal texture synthesis by patch-based sampling and morphing interpolation. *Computer Animation and Virtual Worlds*, 18(4-5):415–428.

Langer, M. S. and Mann, R. (2003). Optical snow. *International Journal of Computer Vision*, 55(1):55–71.

Lockyer, M., Bartram, L., and Riecke, B. (2011). Simple motion textures for ambient affect. In *Proceedings of the International Symposium on Computational Aesthetics in Graphics, Visualization, and Imaging*, pages 89–96. ACM.

Lu, Z. and Sperling, G. (2001). Three-systems theory of human visual motion perception: review and update. *JOSA A*, 18(9):2331–2370.

Lum, E., Stompel, A., and Ma, K. (2003). Using motion to illustrate static 3d shape-kinetic visualization. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 9(2):115–126.

Nishida, S., Ledgeway, T., and Edwards, M. (1997). Dual multiple-scale processing for motion in the human visual system. *Vision research*, 37(19):2685–2698.

Okabe, M., Anjyor, K., and Onai, R. (2011). Creating fluid animation from a single image using video database. In *Computer Graphics Forum*, volume 30, pages 1973–1982. Wiley Online Library.

Ruiters, R., Schnabel, R., and Klein, R. (2010). Patch-based texture interpolation. *Computer Graphics Forum*, 29(4):1421–1429.

Van Wijk, J. (2002). Image based flow visualization. *ACM Transactions on Graphics (TOG)*, 21(3):745–754.

Ware, C. (2004). *Information visualization: Perception for design*, volume 22. Morgan Kaufmann.

Ware, C. and Bobrow, R. (2004). Motion to support rapid interactive queries on node–link diagrams. *ACM Transactions on Applied Perception (TAP)*, 1(1):3–18.

Ware, C. and Knight, W. (1995). Using visual texture for information display. *ACM Transactions on Graphics (TOG)*, 14(1):3–20.