Analysis/Synthesis Approaches for Creatively Processing Video Signals

Javier Villegas
School of Information: Science, Technology, and Arts
University of Arizona
javier.villegasp@gmail.com

Angus Graeme Forbes
University of Illinois at Chicago
Department of Computer Science
aforbes@uic.edu

ABSTRACT

This paper explores methods for the creative manipulation of video signals and the generation of animations through a process of analysis and synthesis. Our approach involves four distinct steps, and different creative outputs based on video inputs can be obtained by choosing different alternatives at each of the steps. First, we decide which features to extract from an input video sequence. Next, we choose a matching strategy to associate the features between a pair of video frames. Then, we choose a way to interpolate between corresponding features within these frames. Finally, we decide how to render these elements when resynthesizing the signal. We illustrate our approach with a range of different examples, including video manipulation experiments, animations, and real-time multimedia installations.

Categories and Subject Descriptors

I.3.3 [**Computer Graphics**]: Picture/Image Generation; I.4.9 [**Image Processing and Computer Vision**]: Applications; J.5 [**Computer Applications**]: Art and Humanities—*fine arts, performing arts*

General Terms

Algorithms, Design, Experimentation

Keywords

Analysis/Synthesis, resynthesis techniques, video processing, animation, media arts, computer graphics

1. INTRODUCTION

Signal alteration is a well established means for artistic expression in the visual arts. Popular tools such as Photoshop, Instagram, and After Effects enable a user to explore creative effects by, for instance, applying filters to an input image or video. We introduce a powerful strategy for the manipulation of video signals that combines the processes of *analysis* and *synthesis*. After an *analysis* process a signal is represented by a series of elements or features. This representation can be more appropriate than the original for a wide range of applications, including, for example, the compression and transmission of video signals [27], or, as we describe in this paper, this representation can be used to generate new modified instances of the starting signal. In the audio domain, Analysis/Synthesis (hereafter, A/S) strategies have been used extensively in creative applications. The phase vocoder is perhaps the best known A/S audio processing algorithm [8]. With the phase vocoder it is possible to manipulate the duration of a signal and the pitch of a signal independently. Other popular effects include dispersion, robotization, whisperization and automatic tuning [44]. However, although A/S techniques are sometimes used for processing videos, in general there is less of an emphasis on using A/S for creative, real-time techniques on video signals.

In one sense, many effects applied on static images, including mosaicing, pointillism, and other non-photorealistic representations, can be thought of as A/S processes. In these processes, a particular set of features (e.g, regions, lines, or objects) are identified through an *analysis* of the input image. These features are then used to describe new elements, which are then *synthesized* into a modified version of the original image. Thus, despite even potentially extreme modifications, the newly-created, non-photorealistic image nonetheless retains many aspects of the identity of the original input. In extending this technique to video input, a common problem with the A/S techniques, and many other non-photorealistic rendering approaches, is that when frames are analyzed independently the detected features can vary abruptly between consecutive frames. This is due to the nature of the detection algorithm or its sensitivity to noise. These rapid variations create distracting artifacts when the independently synthesized frames are put together in an animation. According to Bénard et al., this issue of temporal coherence has prevented non-photorealistic techniques, or stylized animations, from being more widely adopted for video manipulation [1].

We present a novel approach to the implementation of A/S techniques applied to video signals. Our approach involves matching elements between image pairs, i.e., video frames, and involves constructing video processing techniques over one or more of four distinct stages, each one of which enables different creative decisions to be made. These matchings are not necessarily constrained to the image itself, but instead, as we show in Section 4, can take place in a different do-

MM'14, November 03 - 07 2014, Orlando, FL, USA.

ACM 978-1-4503-3063-3/14/11 $15.00.
http://dx.doi.org/10.1145/2647868.2654944.

main. Moreover, we utilize creative criteria for choosing the matching strategy across image pairs. These criteria may be sub-optimal in some senses, and we show that by selecting unconventional interpolation paths and exploring new rendering techniques we can achieve interesting results that are not based on emulating existing methods. The first step of our approach is the *analysis* of the input image. During this step a series of elements are extracted from the image, each described by a set of parameters. The second step is the *matching* of elements between frames. Again, different criteria can be used to do this matching. The problem of matching elements can be thought of as an assignment problem [5]; although efficient solutions exists for finding assignments that are in some sense optimal, these solutions may not be realizable in real-time, and thus not appropriate for interactive experiences. In this paper, we evaluate the possibilities and limitations of *sub-optimal* matching algorithms that can be implemented in real-time. Once the assignment between elements is defined, the next step is to decide on an *interpolation* trajectory in the parameter space; that is, how the elements from one image are going to become the elements of the other. The final step is the *rendering* of the output image. The output does not neeed to be a faithful representation of the input, but ideally aspects of its identity should be maintained. Many narrative possibilities can be explored in this last step.

The contributions of the paper are as follows:

- We introduce a method for thinking of video processing in a particular set of stages in order to enable a range of creative possibilities (Section 3).
- We adapt signal processing approaches common in the audio domain (such as the FFT vocoder, matching pursuits, and cross-synthesis) to the processing of video sequences (Section 2).
- We introduce techniques for creative video processing, including the use of instantaneous frequency manipulation for image transitions (Section 4.6) and the use of 2D Gaussian decomposition to create fluid morphing between video frames (Section 4.3).
- We show that in some cases, sub-optimal matching and other alternative interpolation strategies lead to creative approaches especially useful for real-time video processing (Section 3.2).
- We introduce example artworks that illustrate how to create computer animations that extend algorithmic art techniques by showing that multiple scales (global and local) can be used to explore simultaneous narratives (Section 4).

In addition to describing details of our four-step A/S approach in Section 3, we introduce a series of example projects that utilize this approach in Section 4. We show how radical modifications in the *synthesis* stage can produce engaging results.

The first two examples explore the use of simple feature vectors that describe points defined solely by their x and y position within the image. The third example presents a dictionary-based method that can be combined with a finite state machine to create animations with micro-narratives at local level. In the fourth example we describe an application

Figure 1: An art installation showing trees created in real-time from the output of a dithering algorithm using the sub-optimal matching strategy. Section 4.1 discusses this project. Image copyright ©Javier Villegas.

of a two-dimensional Gaussian decomposition. The fifth example, similar to the previous example, instead uses Fourier descriptors of the closed contours from a level set representation of a grayscale image. The interaction of video data with different media streams is illustrated in the example presented in section 4.5. Finally, in the last example we present interactive animations that can be created by manipulating the transition of the frequency components of a two-dimensional DFT. We discuss the audience reception of our work in Section 5, as well as possibilities for future work in Section 6.

2. RELATED WORK

Much of our present work is inspired by the impact that A/S audio tools have had in the music community, which have motivated the use of new artificial sonic textures in various music genres. In one of the examples we present here, we use a strategy similar to the audio phase vocoder [8]. We detect the phases of different frequency components in consecutive video frames and then use them to manipulate the instantaneous frequency during the synthesis. We have also created animations where the synthesis objects are detected using a template matching algorithm whose structure resembles dictionary-based methods, such as matching pursuits [18]. We also explore the concept of cross-synthesis [44] applied to moving images, that is, images that are synthesized with information from two different sources.

Different works of figurative computer art have used A/S strategies since the 1960s. Mosaic-like versions of photographs have been created by a variety of artists, including Kenneth Knowlton [19], Chuck Close [7], and Robert Silvers [32]. Golan Levin has used similar strategies to algorithmically modify photographs [20, 21]. Daniel Rozin's work has investigated algorithmic "mirrors," interactive systems that recreate input images from a camera in real-time [2].

A/S strategies are also widely used for the effective compression and coding of images [6], since frequency domain representations of real world images tend to concentrate the energy of the image in few coefficients. In Section 4.6 we explore the use of the frequency domain representation of an image to explore different paths for transitions and instant frequency modifications.

The implementation of A/S on moving sequences has also been used to create "rewriting" video applications. In these applications, the frames of a long video are rearranged to create a new video. Bergler et al. used the audio channel on speech videos to associate "visemes" with phonemes, and then automatically generated a video with a new audio signal via a concatenation of visemes at different temporal positions [4]. Schödl and Essa created a technique, "Video Sprites," that can rearrange and correct the perspective of long video sequences of animals moving freely [30]. Our approach to video resynthesis is somewhat different as we are not simply reorganizing the existing video frames, but rather recreating new ones through resynthesizing and concatenating particular components of the video.

Non-photorealistic rendering (NPR) techniques investigate the automatic recreation of different styles of hand painting. Hetzmann unified many of those techniques under the term "stroke-based rendering" [15]. In his survey, Hertszmann describes the general problem as designing an optimization algorithm that minimizes the placement error of different types of strokes. Similarly, Gooch et al. [13] illustrate a technique with two clear steps: first, finding the best position for strokes and, second, rendering those brush strokes on the virtual canvas. Other authors have used techniques where an A/S process is more evident. For instance, Li and Huang [22], Hausner [14], and Wen et al. [42] discuss the detection of regions to be used as a stroke container or as a render primitive, and Lin et al. [23] uses a combinatorial matching algorithm to track the propagation of strokes. In our work we also suggest the use of combinatorial optimization to find correspondences between the elements of pairs of frames, but we propose the use of different matching optimization criteria as an important part of the creative control. A recent review of different NPR techniques and applications can be found in [29].

Gooch et al. emphatically state that the NPR community has surpassed the stage of mimicking art styles from the past and should instead be looking forward to generating novel rendering styles that are only possible with computers [12]. We are also interested in creating visual experiences that promote interaction and that can be implemented on mobile devices. The computational power and hardware capabilities of today's mobile devices are an invitation to use them as the destination platform for real-time A/S based manipulations. Many contemporary artists have chosen to create artworks for mobile platforms [9, 16, 33, 41].

Our approach can also be used to create transitions between two dissimilar images. Many morphing techniques have been created to generate fluid transitions between pairs of images [43]. In most of these techniques, there is a tradeoff between the complexity of the definition of the correspondent points and the warping function. With our system, new smooth transitions between pairs of images can be generated even when it is not possible to define correspondent points by visual similarity.

We present a variety of rendering primitives that are *not* based on any form of painting or illustration (a common approach of NPR techniques [15]) and that can moreover be used with the combinatorial optimization matching algorithms we present here to create animations and video effects. Furthermore, we also present sub-optimal alternatives that can be used effectively in interactive applications, including on mobile devices.

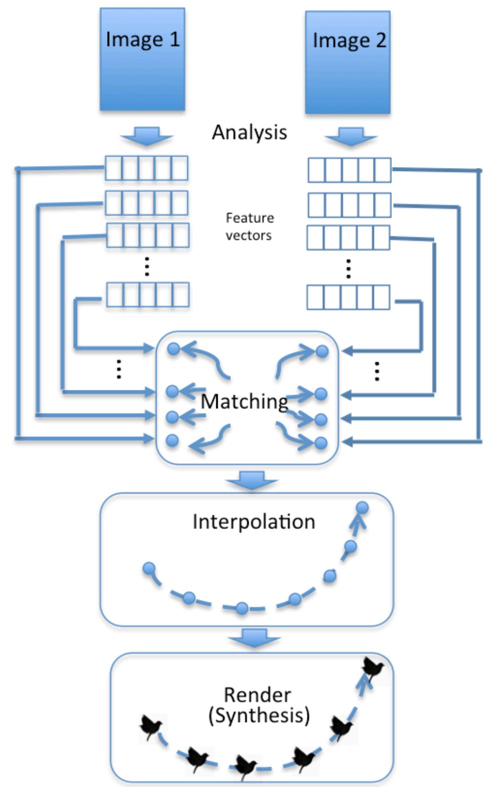


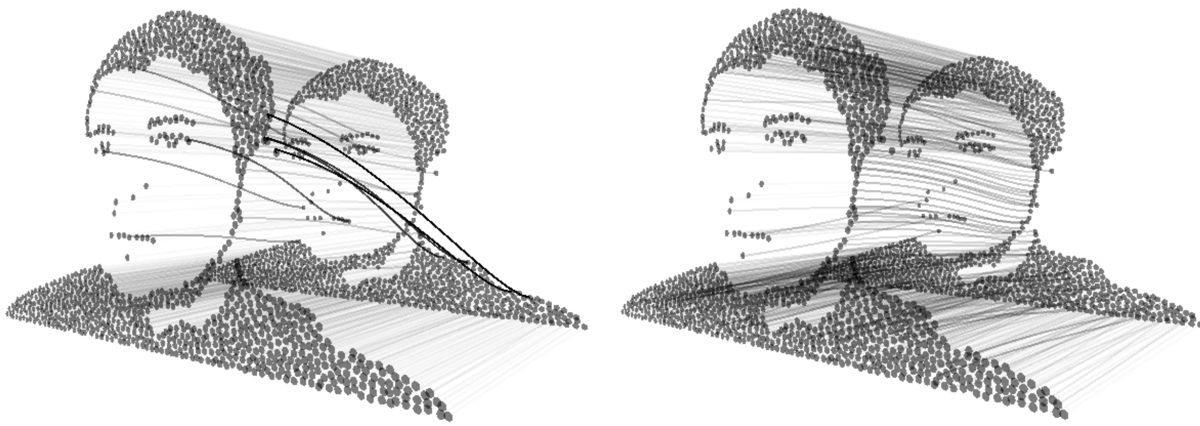
Figure 2: An overview diagram showing the four stages of the system. For step 1, we determine which feature to extract (and how to extract them) from the video sequence. For step 2, we choose a matching strategy to associate specific features between image pairs. For step 3, we decide how to interpolate between corresponding features. Finally, in step 4, we choose a rendering technique that resynthesizes the features and places them in an output video sequence. Creative choices can be made at each of these four stages.

3. GENERAL DESCRIPTION

An overview of our four-step approach to A/S is depicted in Figure 2. Pairs of images are analyzed to extract descriptive elements from them. These elements are then matched and the transitions between the correspondent pairs are calculated. In the final stage a new image is created by rendering the modified elements back to screen.

3.1 Analyzing the Input Image

The first step in the process is the *analysis* of the image. By definition, an analysis is the process of separating a whole into its component parts, but many different alternatives can be considered as "component parts." We interpret this stage



(a) (b)

Figure 3: These figure depict different matching criteria between frames. In (a), two consecutive frames and the lines joining the matched objects. The darkness of the lines is proportional to the distance. In (b), the same two frames as the previous example but with matchings defined using the minimum maximum criterion. Here, we see that it is less likely that any one match will jump a significant distance.

broadly, considering it as a process of identifying and extracting information that can then be used in later steps. We explore a range of possible outputs from this stage, including: the decomposition of the image into a transformed domain, such as the DFT [11]; or the list of features from a feature detection algorithm (e.g., black points in a dithered image, straight lines from the edge map, connected regions, or corners) [25]. After this process the input image is represented as a set of points in the transformed domain, that is, in the space formed by the parameters of the detected features.

3.2 Matching Elements between Frames

3.2.1 Optimal Matching

The features on each frame are matched with the features of the next frame. If the number of detected features is the same on all frames then the matching can be posted as an assignment problem as follows:

- Define the cost matrix $\mathbf{C}[n, n+1]$ of size $M \times M$ where M is the number of features on each frames, and each matrix element C_{ij} represents the Euclidean distance between the vector of parameters $\mathbf{z}$ of every feature i on frame n to every feature j on frame $n+1$. That is,

$$\mathbf{C}_{i,j} = d\left(\mathbf{z}_i[n], \mathbf{z}_j[n+1]\right). \tag{1}$$

- We then want to find the assignment matrix $\mathbf{X}[n]$ with elements X_{ij}, where $X_{ij} = 1$ if the feature i on frame n is matched to feature j on frame $n+1$ and $X_{ij} = 0$ otherwise.

- The assignment matrix is restricted so that each feature on frame n is assigned to one and only one feature on frame $n+1$ and reciprocally each feature on frame $n+1$ is matched to one and only one feature on frame n. That is,

$$\sum_{j=1}^{M} X_{ij} = 1 \qquad (i = 1.2...M)$$

and,

$$\sum_{i=1}^{M} X_{ij} = 1 \qquad (j = 1.2...M)$$

with,

$$X_{ij} \in \{0, 1\} \qquad (i, j = 1.2...M)$$

- If the assignment matrix that is chosen is the one that minimizes the sum of distances between features, that is, if the matrix $\mathbf{X}[n]$ is chosen such that

$$\sum_{i=1}^{M} \sum_{j=1}^{M} C_{ij} X_{ij} \tag{2}$$

is minimized, the optimization problem is known as the linear assignment problem, and many algorithms for solving this problem efficiently can be found in the literature [5]. But on the other hand, if the matching criterion is not to minimize the sum of distances but to make the maximum distance as small as possible, the assignment problem is transformed to the bottleneck assignment problem. The problem now is to minimize the maximum of $C_{ij}X_{ij}$ $\forall i, j = 1 \cdots M$ with the same constrains presented previously. This is a very well known variation of the original assignment problem for which, fortunately, efficient algorithms also exist [5].

Figure 3a shows two frames of an animation where, for simplicity of illustration, the synthesis objects are circles with fixed color and size. Every feature can be described by a vector of only two parameters (the x and y position) and the matching between the objects is done using the minimum sum criterion. Larger distances are plotted with a darker tone. Note that in order to minimize the total sum, most of the matchings are done between near objects. Figure 3b shows the result of the bottleneck assignment problem with the same two frames. Note that, though many matches have significant distances (darker lines), there are no big jumps.

3.2.2 *Sub-optimal strategies*

Although some of the limitations of the combinatorial optimization approaches, such as the restriction on having the same number of elements, can be easily overcome (for instance by repeating elements on the less populated images), they are nonetheless not feasible for real-time implementation. We explore a sub-optimal alternative to the matching of elements that consists of these following steps:

- A maximum number of elements N is defined so that the number of detected features in the analysis stage is never larger than this number;
- At the beginning, N random elements are created;
- Starting with an arbitrary element, each one of them is assigned to the closest (in the parameter space) unassigned feature in the currently analyzed frame;
- If all the target features are assigned, the closest one with less source elements associated to it is chosen.

Despite the simplicity of this alternative and its sensitivity to the starting element, we have obtained good results using this algorithm in real-time interactive experiences. Some of these results are described in section 4.1.

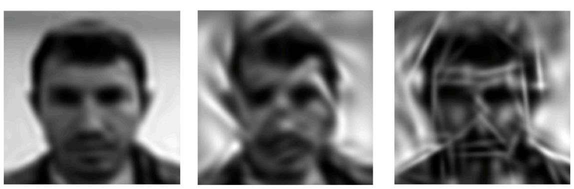


Figure 4: Using different paths to get to the same target image. The left image is the final reconstruction with a linear combination of two-dimensional Gaussian functions. In the middle image the last parameter to be updated was the angle. In the right image the last parameter was the variance.

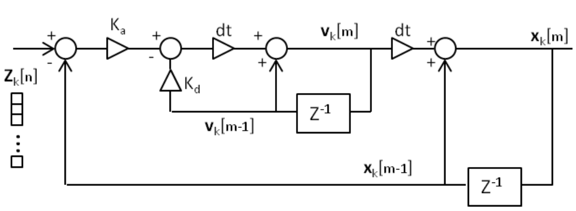


Figure 5: Block diagram of an Euler integrator used to generate a transition trajectory. $Z_k[n]$ is the current detected feature, $X_k[n]$ is the interpolated output.

3.3 Creating Intermediate Images by Interpolation

Once the matching between elements has been decided, the next aspect to be determined is the path in parameter space that each source element will follow to become its correspondent target element. One of the alternatives that we have used is illustrated in Figure 5. The vector of the difference between the current and desired position is used as a steering force, and then the velocity and position of the elements in parameter space is calculated by discrete time integration. The two parameters K_a and K_d control the degree of influence of the steering and drag forces and can be modified to generate different trajectories.

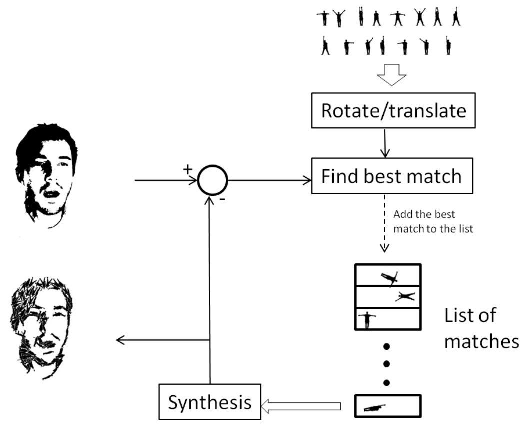


Figure 6: A dictionary-based strategy to detect different silhouettes in a black and white image.

Figure 7: The frame of an animation created with the dictionary-based scheme. Every small element is a human silhouette in a different pose. Image copyright ©Javier Villegas.

In many of the representations that are obtained after the analysis stage, the visual salience of the parameters that conform each element is not necessarily uniform (e.g., in the Fourier representation of images, phase is much more important than amplitude [11]). Therefore, perceptually different transitions can be obtained if the parameters of each element do not change in perfect synchronicity (See Figure 4).

Table 1: Summary of steps used in example projects

Project	Analysis	Features	Matching	Interpolation	Synthesis	Interactive?
Ant Theater (Section 4.1)	Corner detection	2D points	Sub-optimal	Euler integration	Ant textures	Yes
Herbaceous (Section 4.1)	Dithering	2D points	Sub-optimal	Euler integration	Tree structures	Yes
Background Singer (Section 4.2)	Dictionary of silhouette templates	2D oriented elements with state	Minimum maximum	Euler Integration; Viterbi	Human silhouettes	No
Untitled (Section 4.3)	Nonlinear least squares	2D Gaussian functions (6D vectors)	Minimum sum	Euler integration	2D Gaussian functions	No
Untitled (Section 4.4)	Connected regions; Contour detection	128D Fourier descriptors	Minimum sum	Euler integration	Filled regions	No
Slave of Your Words (Section 4.5)	Luminance value	Rectangular regions	Identity; One on one	Zero-order hold	Audio waveforms	Yes
The Fitting Dance (Section 4.5)	Ellipse Fitting	Ellipses	Sub-optimal	Euler integration	Ellipses w/audio-controlled contours	Yes
Untitled (Section 4.6)	2D FFT	256x256 coefficient matrix	Identity; One on one	Asymmetric frequency-dependent Euler integration	Frequency modulation of the IFFT	Yes

In section 4.6 we illustrate these possibilities with specific examples.

3.4 Resynthesizing the Output Image

The final step is to recreate a new image, that is, to render the elements that have been interpolated previously. It is in this step where narrative meanings can be injected into the animation, for example, by replacing corners detected in the original frame with new images related to a particular theme (Section 4.1), or by introducing visual motifs taken from other modalities (Section 4.5). This can allow the artist to emphasis an aspect of the original video or reveal a new story superimposed upon the original video. As we will illustrate in Section 4, the elements drawn on screen do not have to be exactly the same ones that were detected, nor do they need to be abstract entities. Rather they can have an identity of their own, based perhaps on the aesthetic goals of the artwork. They can also be used as a starting point or as input parameters for a more sophisticated final rendering process.

4. EXAMPLES

In this section we will present examples of animations and real-time installations that follow the approach presented above. Table 1 summarizes the characteristics of each of the examples, illustrating how our method enables a variety of creative techniques at various stages of our A/S approach, including feature extraction, matching, interpolation, and rendering. We also indicate which techniques are appropriate for interactive contexts; in general, techniques that use a sub-optimal matching strategy can occur at real-time rates.

Figure 8: An "interactive mirror" using the sub-optimal algorithm and rendering points as ants.

4.1 Point-like Features

We developed a real-time installation, *Ant Theater* [37], where an input image is analyzed in real-time using a Shi-Tomasi corner detection algorithm [31] (see Figure 8). The assignment between frames is done using the sub-optimal algorithm described in Section 3.2.2. To handle the different number of elements detected on each frame, a maximum number of elements is predefined (it is a number that is always bigger than the maximum number of corners that the algorithm can return). Some elements are matched to to the same target point, so at render time some ants will be simply hiding behind others. Intermediate positions are calculated using the Euler integration scheme presented in section 3.3. The features are the points detected as corners but they are rendered as ants.

A similar approach was used for the real-time installation, *Herbaceous*, shown in Figure 1 [39]. Here the analysis stage instead uses a Floyd-Steinberg dithering algorithm [34] modified to return a restricted maximum number of points (the points that represent darker areas). The points are again two dimensional features that are matched using the sub-optimal approach and then rendered as the leaves of a tree. The body of the tree is constructed using the position of the leaves, as suggested by Rodkaew et al. [28].

4.2 Template Matching

Figure 7 shows a frame from an animation where the analysis stage involved the detection of human silhouettes in different positions and orientations. This scheme, depicted in Figure 6, resembles dictionary-based decompositions such as the matching pursuits algorithm [18]. Matching between frames is done using the minimum maximum optimization criterion. Elements in this representation have position, orientation and state. Position and orientation can be interpolated with the strategy depicted in Figure 5, but for the interpolation of states, a variation of the Viterbi algorithm was used to generate the intermediate states [24]. This technique was used for the creation of a short animation that explored the possibilities of narratives a two different levels [38].

4.3 Gaussians in a 6D Space

Figure 11 shows the morphing between two images using features that belong to a six dimensional space. The features are Gaussian functions. The target image was analyzed using non-linear least squares [26] to find the best fit of a linear combination of 2D Gaussian functions. That is, a grayscale image is approximated by:

$$I\left(x,y\right) \approx \sum_{n=1}^{N} g_n\left(x,y\right) \tag{3}$$

where

$$g_n\left(x,y\right) = A_n e^{-a_n\left(x-\mu_{xn}\right)^2 + 2b_n\left(x-\mu_{xn}\right)\left(y-\mu_{yn}\right) + c_n\left(y-\mu_{yn}\right)^2}$$

with

$$a_n = \frac{\cos^2\theta_n}{2\sigma_{xn}^2} + \frac{\sin^2\theta_n}{2\sigma_{yn}^2}$$

$$b_n = -\frac{\sin 2\theta_n}{4\sigma_{xn}^2} + \frac{\sin 2\theta_n}{4\sigma_{yn}^2}$$

$$c_n = \frac{\sin^2\theta_n}{2\sigma_{xn}^2} + \frac{\cos^2\theta_n}{2\sigma_{yn}^2}$$

Thus, each Gaussian function can be represented as a point in a six-dimensional space using the six parameters: A_n, μ_{xn}, μ_{yn}, σ_{xn}, σ_{yn}, θ_n.

In the sequence depicted in Figure 11, 144 Gaussian functions are used to represent each one of the images. The Gaussian functions are then matched using the minimum sum criterion, and the parameters from the source image are slowly transformed to the destination image.

4.4 Fourier Descriptors in a 128D Space

A similar experiment in morphing two images can be repeated with a different set of features. This time we used the set of closed contours of a level set representation of a grayscale image (see Figure 9).

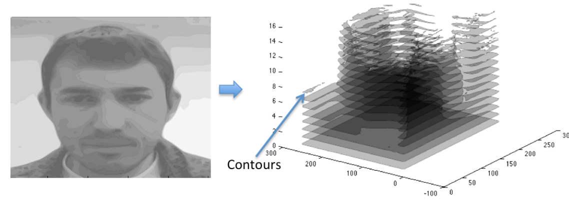


Figure 9: Generating a list of the level sets of a grayscale image.

Each contour is represented using a 128-length vector of its Fourier descriptors [17]. The list of contours of the original image is matched using the minimum sum criterion with the list of contours of the destination image. To calculate the cost matrix $\mathbf{C}[n, n+1]$ of equation 1, we used the Euclidean distance between the magnitude of all but the first element of each vector of Fourier descriptors. By doing that we consider differences in shape, scale and position and ignore differences in starting point and (unfortunately) orientation. Figure 13 shows a sequence of frames created in the morphing between the two representations.

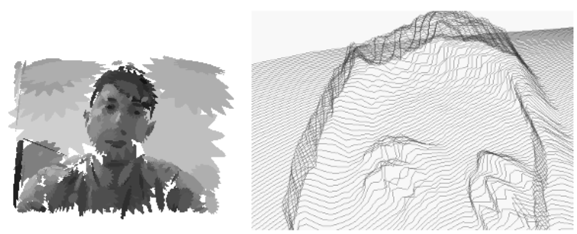


Figure 10: Two images showing the use of an audio signal before the resynthesis. On the left the RMS value of the audio signal is used to change the contours of the ellipses. In the picture on the right the waveform of the audio signal is used to draw the input image.

4.5 Intermodality

A/S approaches are powerful in part because strong manipulations can be performed before rebuilding the output. Figure 10 shows examples where an external signal (i.e. an audio stream) is used to transform the input image. The left image shows an image where the RMS value of a musical input is used to determine the amplitude of the sinusoidal signal added to the ellipse contours. This strategy was used for the creation of a short animation titled *The Fitting Dance* [35]. The right image shows a zoom in to a frame of the interactive installation *Slave of Your Words* [36]. In this installation the waveforms themselves are used as synthesis elements.

4.6 A/S in the Frequency Domain

The Fourier transform is a powerful analytical tool commonly used for signal transformations [3]. For this last example, we use the two-dimensional Fourier transform in the

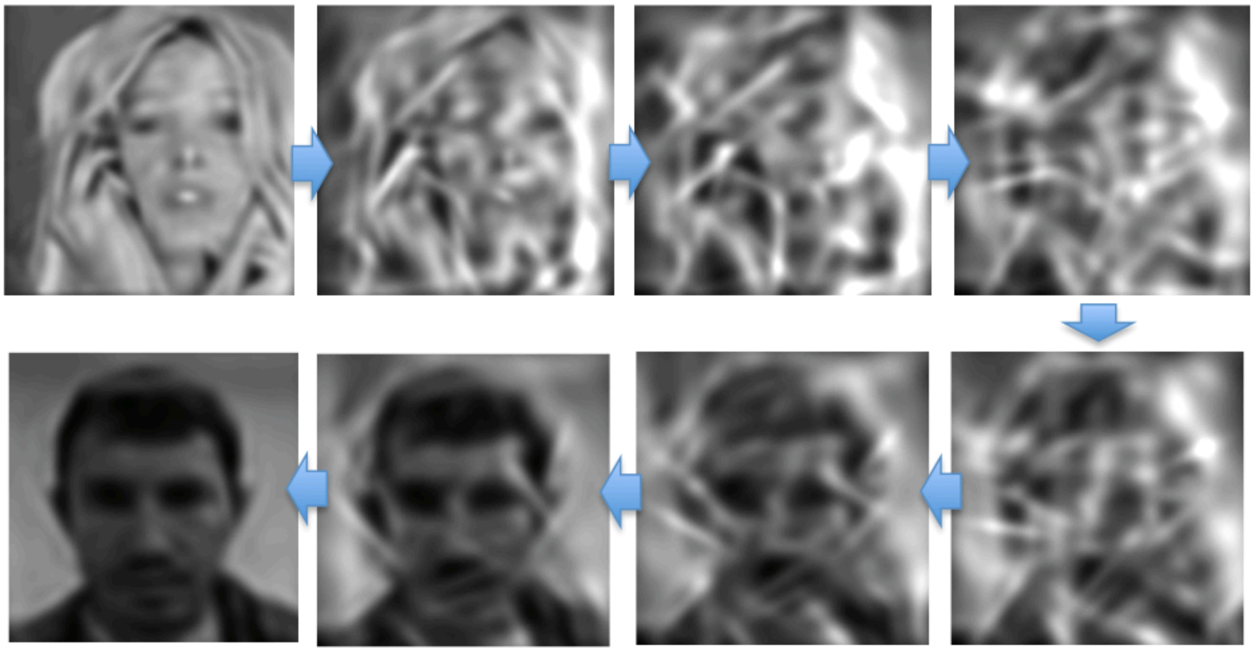


Figure 11: Morphing between two images using a six-dimensional representation for each of the detected feature (a 2D gaussian function), and matching them with a minimum sum criterion.

Figure 12: A frame of the FFT animation. The right side shows the resynthesized version created by frequency modulation of the IFFT of the phase interpolated version of the input. The middle image shows that low frequencies are updated first.

analysis stage. After this process, the image is represented as a linear combination of two-dimensional sinusoidal functions of different phases and amplitudes. We used a trivial identity assignment of the frequency components from frame to frame, but we used the interpolation scheme shown in Figure 5 for the phases of the low-frequency detected components. The high frequency components are updated only after the low frequency elements are close enough to their targets. The resulting animation is rendered as the frequency modulation of the inverse Fourier transform of the interpolated components. To interact with the piece, the phase of the low frequency components can be manipulated in real time with dragging gestures. This example is illustrated in Figure 12.

5. EVALUATION

In this paper, we presented a novel approach for thinking about how to generate moving sequences from video data. We have used this approach to create a wide range of animations and real-time installations that have been received positively by audiences and curators.

In many of the applications we have created, the rendered elements are constantly moving towards a target. Because of that, a faithful reconstruction of the input is accomplished only when the image stays still for a few frames. That constraint defines what type of sequences work best as subject matter. For instance, we have used slow moving faces in animations like *The Fitting Dance* [35] and *Background Singer* [38], both of them were presented in multiple international venues, including the 2009 Japan Media Arts Festival[1] (more than 10,000 attendees), the 2011 Byte Gallery International Exhibition[2], and the 2012 Portland Experimental Film Festival[3] (each with ∼1,000 visitors). Audience reaction was enthusiastic as the artworks engaged the audience with a narrative in which the synthesis elements changed with time according to a backstory. There is always a mixture of joy and surprise when familiar objects (like human faces) are created slowly from the synthesis objects.

In real-time installations we have found that, rather than being a limitation, the non-immediate response of the system in fact adds suspense. It becomes a way to engage the viewer as a participant and to invite him or her to spend some time with the artwork. We have observed this behavior in some of the more popular installations, such as *Herbaceous* [39] and *Ant Theater* [37]. Both of these pieces have been presented in international exhibitions, including the 2014 Digital Latin America Festival[4] (still ongoing, ∼5,000 attendees expected), Currents 2013: The Santa Fe International New Media Festival[5] (>5,000 attendees), and the 2012

[1] `http://j-mediaarts.jp/about/index?locale=en`
[2] `http://www.transy.edu/music/BYTE_GALLERY`
[3] `http://effportland.com/`
[4] `http://issuu.com/516artsabq/docs/digital_latin_america_program_guide`
[5] `http://currentsnewmedia.org/`

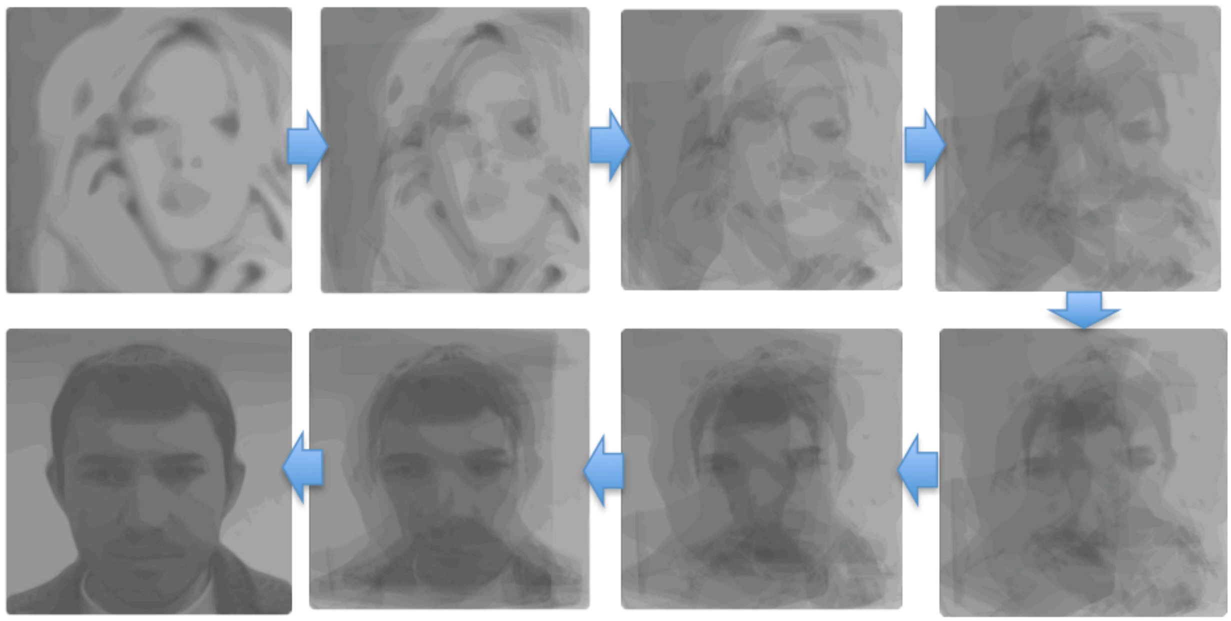


Figure 13: Features of dimension 128 matched between two images and the intermediate images that they generate.

Prospectives International Festival of Digital Art[6] (∼1,000 attendees).

6. FUTURE WORK

We have shown that it is possible to use our approach to create A/S techniques for video sequences similar in many ways to A/S techniques in the audio domain. Despite the perceptual differences between aural and visual stimuli, the most successful audio tools can still be used as inspiration to explore the possibilities of A/S in animation. In particular, we showed how the two-dimensional Fourier transform can be used to create a continuous manipulation of frequency components at interactive rates, similar to the way the phase vocoder does for the manipulation of audio partials. Future work can include the use of an oscillator bank in the resynthesis stage instead of the IFFT. That implementation will allow more direct manipulation of frequency, orientation, phase, and amplitude before the resynthesis step.

We strongly believe that the use of A/S techniques on video sequences can support the creation of new visual narratives. For example, we showed in section 4.1 that the rendering step can be used to impose an additional "local" meaning to the elements to be rendered. This simultaneity of subject matters at multiple scales can be used to explore new parallel narratives [40].

We have shown that our approach is general enough to be adequate for the creation of non-photorealistic representations that are not intended to mimic art styles from the past, but instead seek to find novel creative renditions of moving images. Although some of the alternatives we presented here are currently not suitable for real-time implementation, we expect that in the near future that situation will be overcome.

[6] `www.unr.edu/art/prospectives12.html`

We have created animations [38, 35], real-time installations [10, 37, 39, 36] and tablet applications [41] with our techniques, but an interesting direction to distribute and further evaluate the creative potential of our work will be to create public domain tools that allow other people to experiment with our techniques. Although not discussed here in this text, future work will explore the creation of plugins and interfaces that help artists to better make use of A/S techniques, such as those described above. Additionally, we plan to introduce authoring tools that enable multimedia programmers to more easily develop and test novel A/S techniques appropriate for their own projects.

7. REFERENCES

[1] P. Bénard, A. Bousseau, and J. Thollot. State-of-the-art report on temporal coherence for stylized animations. In *Computer Graphics Forum*, volume 30, pages 2367–2386. Wiley Online Library, 2011.

[2] J. D. Bolter and D. Gromala. *Windows and Mirrors: Interaction Design, Digital Art, and the Myth of Transparency*. The MIT Press, Oct. 2005.

[3] R. N. Bracewell. The Fourier transform and its applications. 1980.

[4] C. Bregler, M. Covell, and M. Slaney. Video rewrite: Driving visual speech with audio. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 353–360. ACM Press/Addison-Wesley Publishing Co., 1997.

[5] R. Burkard, M. Dell'Amico, and S. Martello. *Assignment problems*. Society for Industrial Mathematics, 2009.

[6] R. J. Clarke. Transform coding of images. *Astrophysics*, 1, 1985.

[7] C. Finch. *Chuck Close: Work*. Prestel Publishing, Oct. 2007.

[8] J. L. Flanagan and R. Golden. Phase vocoder. *Bell System Technical Journal*, 45(9):1493–1509, 1966.

[9] A. G. Forbes, T. Höllerer, and G. Legrady. Generative fluid profiles for interactive media arts projects. In *Proceedings of the International Symposium on Computational Aesthetics in Graphics, Visualization, and Imaging*, pages 37–43, 2013.

[10] A. G. Forbes and J. Villegas. Creative applications of microvideos. In *MMEDIA 2014, The Sixth International Conferences on Advances in Multimedia*, pages 108–111, 2014.

[11] R. C. Gonzalez and R. E. Woods. Digital image processing, 2nd. *SL: Prentice Hall*, 2002.

[12] A. A. Gooch, J. Long, L. Ji, A. Estey, and B. S. Gooch. Viewing progress in non-photorealistic rendering through heinlein's lens. In *Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering*, pages 165–171. ACM, 2010.

[13] B. Gooch, G. Coombe, and P. Shirley. Artistic vision: painterly rendering using computer vision techniques. In *Proceedings of the 2nd international symposium on Non-photorealistic animation and rendering*, pages 1–11. ACM, 2002.

[14] A. Hausner. Simulating decorative mosaics. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 573–580. ACM, 2001.

[15] A. Hertzmann. A survey of stroke-based rendering. *IEEE Computer Graphics and Applications*, 23(4):70–81, 2003.

[16] L. Jevbratt. Zoomorph. `http://zoomorph.net`, 2011.

[17] B. Jähne. *Digital Image Processing 6th Edition.* Springer, Berlin [u.a.], 2005.

[18] G. Kling and C. Roads. Audio analysis, visualization, and transformation with the matching pursuit algorithm. In *Proceedings of the 7th International Conference on Digital Audio Effects (DAFx-04)*, pages 33–37, 2004.

[19] Knowlton. Knowlton mosaics - portraits by computer assisted art pioneer Ken Knowlton. `http://www.knowltonmosaics.com`, Accessed July 2014.

[20] G. Levin. Floccular portraits - interactive art by Golan Levin and collaborators. `http://www.flong.com/projects/floccugraph`, Accessed July 2014.

[21] G. Levin. Segmentation and symptom - interactive art by Golan Levin and collaborators, Accessed July 2014.

[22] N. Li and Z. Huang. A feature-based pencil drawing method. In *Proceedings of the 1st international conference on Computer graphics and interactive techniques in Australasia and South East Asia*, pages 135–ff. ACM, 2003.

[23] L. Lin, K. Zeng, H. Lv, Y. Wang, Y. Xu, and S.-C. Zhu. Painterly animation using video semantics and feature correspondence. In *Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering*, pages 73–80. ACM, 2010.

[24] H. Lou. Implementing the viterbi algorithm. *Signal Processing Magazine, IEEE*, 12(5):42–52, 1995.

[25] M. S. Nixon and A. S. Aguado. *Feature Extraction & Image Processing for Computer Vision.* Academic Press, 2012.

[26] J. Nocedal and S. J. Wright. *Numerical Optimization.* Springer, New York, 2nd edition, 2006.

[27] C. Poynton. *Digital video and HD: Algorithms and Interfaces.* Elsevier, 2012.

[28] Y. Rodkaew, P. Chongstitvatana, S. Siripant, and C. Lursinsap. Particle systems for plant modeling. *Plant Growth Modeling and Applications*, pages 210–217, 2003.

[29] P. Rosin and J. Collomosse. *Image and Video-Based Artistic Stylisation*, volume 42. Springer, 2013.

[30] A. Schödl and I. A. Essa. Controlled animation of video sprites. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 121–127. ACM, 2002.

[31] J. Shi and C. Tomasi. Good features to track. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 593–600, 1994.

[32] R. Silvers. Robert silvers. `http://www.photomosaic.com`, 2003.

[33] S. Snibbe. Studio apps. `http://www.snibbestudio.com/apps`, accessed July 2014.

[34] T. Strothotte and S. Schlechtweg. *Non-Photorealistic Computer Graphics: Modeling, Rendering and Animation.* Morgan Kaufmann, 1 edition, June 15 2002. ISBN: 1558607870, pages 472.

[35] J. Villegas. The fitting dance - animation, 2009. `https://vimeo.com/12137723`.

[36] J. Villegas. Slave of your words - interactive installation, 2011. `https://vimeo.com/31750240`.

[37] J. Villegas. Ant theater - interactive installation, 2012. `https://vimeo.com/74593738`.

[38] J. Villegas. Background singer - animation, 2012. `https://vimeo.com/43152993`.

[39] J. Villegas. Herbaceous - interactive installation, 2012. `https://vimeo.com/55131863`.

[40] J. Villegas and A. Forbes. Double-meaning: Interactive animations with simultaneous global and local narrative. In *Proceedings of the Re-new Digital Arts Festival*, pages 302–306, October 2013.

[41] J. Villegas and A. Forbes. Interactive non-photorealistic video synthesis for artistic user experience on mobile devices. In *Proceedings of the 8th International Workshop on Video Processing and Quality Metrics for Consumer Electronics (VPQM)*, Chandler, Arizona, February 2014.

[42] F. Wen, Q. Luan, L. Liang, Y.-Q. Xu, and H.-Y. Shum. Color sketch generation. In *Proceedings of the 4th international symposium on Non-photorealistic animation and rendering*, pages 47–54. ACM, 2006.

[43] G. Wolberg. Image morphing: A survey. *The visual computer*, 14(8):360–372, 1998.

[44] U. Zölzer. *DAFX: Digital Audio Effects.* Wiley, 2011.