Revealing Information by Averaging

SAMI ARPA[1*], SABINE SÜSSTRUNK[1], AND ROGER D. HERSCH[1]

[1] School of Computer and Communication Sciences, Ecole Polytechnique Federale de Lausanne
[*] Corresponding author: sami.arpa@epfl.ch

Compiled February 27, 2017

We present a method for hiding images in synthetic videos and reveal them by temporal averaging. The main challenge is to develop a visual masking method that hides the input image both spatially and temporally. Our masking approach consists of temporal and spatial pixel by pixel variations of the frequency band coefficients representing the image to be hidden. These variations ensure that the target image remains invisible both in the spatial and the temporal domains. In addition, by applying a temporal masking function derived from a dither matrix, we allow the video to carry a visible message that is different from the hidden image. The image hidden in the video can be revealed by software averaging, or with a camera, by long exposure photography. The presented work may find applications in the secure transmission of digital information.

OCIS codes: *330.1800 Vision - contrast sensitivity; 330.6790 Temporal discrimination; 330.4595 Optical effects on vision*

http://dx.doi.org/10.1364/ao.XX.XXXXXX

1. INTRODUCTION

The goal of the present work is to hide an image in a video stream under the constraint that the temporal average of the video reveals the image. Specifically, the input image should remain invisible in each frame of the video and should not become visible due to the temporal integration of consecutive frames by the human visual system (HVS). In order to achieve this, a visual masking method that acts both in the spatial and in the temporal domain is required. Spatial masking inhibits orientation and frequency channels of the HVS. In temporal masking, any information coming from the target image by temporal averaging should be masked. As surveyed by Chandler [1], many methods of masking have been presented for the spatial domain. However, masking in the temporal domain has not received much attention.

Our algorithm hides an input image within a video. The image is revealed by averaging, which is either achieved by pixelwise mathematical averaging of the video frames or by long exposure photography (Fig. 1). We call the video hiding the input image *tempocode*.

In order to create such tempocodes, we apply the following self masking model. We first decrease the dynamic range of the input image and decompose it into a certain number of frequency bands [2]. For each frequency band of the contrast reduced input image, we generate temporal samples by sampling a selected masking function, whose integration along a certain time interval gives the corresponding frequency band. We then reconstruct each video frame from the temporal samples derived from the frequency bands. We consider the following masking functions: random function, sinusoidal composite wave function, and a temporally-varying dither function. Using these functions we generate different masking effects such as smoothly evolving videos and videos with visible moving patterns.

Fig. 1. A tempocode is a video (left) containing a hidden image that can be revealed through long exposure photography of the video (right).

Contribution. By exploiting the limitations of the human visual system with respect to the temporal domain, we design an algorithm for creating special video seals. Such a synthesized video either appears as spatial noise or carries a visible message different from the hidden one. If the correct exposure time is set, the hidden image is revealed by a camera. Hiding an image in a video can be used as a security component within a framework for transmitting and authenticating electronic documents, e.g., electronic tickets or digital identity cards.

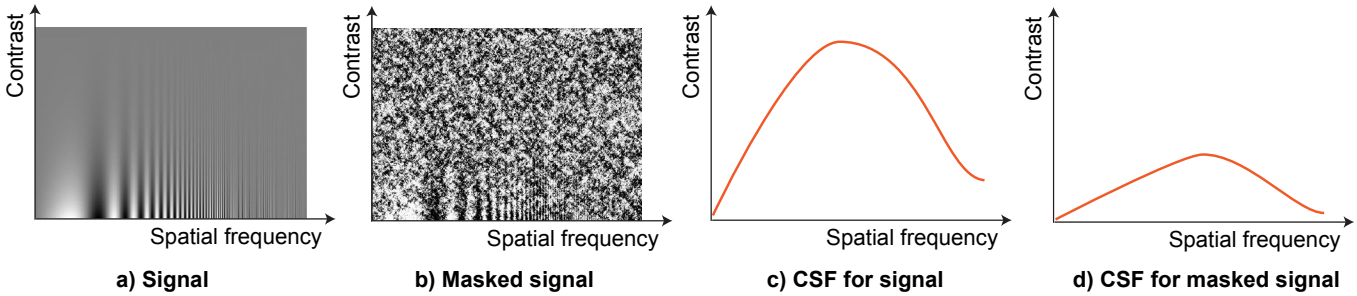


Fig. 2. Masking thresholds are proportional to the contrast sensitivity function (CSF). a) The Campbell-Robson chart [3] indicates the shape of CSF function in an intuitive way. b) Our masking function (Sec. 5A) is applied to the Campbell-Robson chart. Due to masking, the visibility thresholds are reduced. This reduction (d) is fairly proportional to the contrast sensitivity function (c).

2. BACKGROUND AND RELATED WORK

We need to ensure the invisibility of the target image both spatially and temporally. For spatial invisibility, e.g., invisibility within a single frame, we utilize visual masking methods. For temporal invisibility, e.g., when the video is playing, the temporal integration of the HVS is taken into account. Let us first describe spatial visual masking and then explain the temporal integration of the HVS.

Visual masking. The presence of a mask can reduce the ability of a subject to detect a given target signal. This phenomenon is called visual masking [4]. A *masking threshold* is the limit of the contrast of a target for a given contrast of a mask. If the contrast of the target is higher than this threshold, masking is not achieved [1]. In classical image rendering, one tries to mask the artifacts caused by operations such as compression, quantization, and dithering.

Visual masking is still not well-understood. It is observed that early channels of the HVS, where the data is processed at low level, and high level mechanisms of the HVS, where object perception is performed, exhibit different masking behaviours. For instance, masks having unstructured and structured shapes, respectively, result in completely different *masking thresholds* [5].

Masking is the strongest between stimuli located in the same perceptual channel. This is true for both early channels, e.g., the stimuli having the same spatial frequencies and orientations (Fig. 3), and for higher level mechanisms, e.g., stimuli having similar shape structures.

In order to implement visual masking, one generally needs a *masking threshold* function. A target signal can be masked with a mask signal by applying a contrast reduction on the target signal until the *masking threshold* is reached. As shown by Daly [6], for low contrast masks, there is no masking. For high contrast levels, as the mask contrast increases, the *masking threshold* increases as well in a non-linear fashion. However, an unstructured mask, e.g., a noise mask, and a structurally coherent mask, e.g., a sinusoidal grating, have different *masking threshold* functions.

The *masking thresholds* also vary for different spatial frequencies and orientations. The human visual system exhibits different responses for a signal of the same contrast at different spatial frequencies and orientations. These responses are defined by a *contrast sensitivity function* (*csf*). Fig. 2 shows that the *csf* of the masked signal (Figs. 2b, 2d) is approximately proportional to the *csf* of the unmasked signal (Figs. 2a, 2c).

In order to explain visual masking, many other paradigms such as noise masking [8], pattern masking [9], and entropy masking [10] have been presented. Our approach is inspired by Daly's *masking threshold* function. We reduce the contrast of the target in order to reach the *masking threshold*.

Fig. 3. Visual masking. A uniformly laid out distortion is added to the reference image (left). The distortions are invisible on the vertical stripes of the zebra (right). Courtesy [7].

Temporal integration of the human eye. Temporal integration is related to the ability of the HVS to sum the quantity of light exposed over time [11]. According to Bloch's law, the perceived luminous energy is a constant formed by the product of luminance and exposing time. For instance, the perceived luminance of a signal does not change if the intensity of the signal is halved but the exposing time is doubled [12]. However, temporal integration might occur only until a critical duration. This critical duration is approximately $40 \pm 10ms$ [11]. It depends on many factors such as spatial resolution and luminance adaptation according to photopic or scotopic vision. When the exposing time is longer than the critical duration, Bloch's law does not hold and the perceived luminance only depends on the intensity of the signal.

With respect to our work, it is important that the averaging behaviour of the HVS does not reveal the parts of the target image that are to be hidden.

Steganography. Steganography is a technique used for secret communication. The secret message is not necessarily related to the visible content. Furthermore, the secret information is more important than visible content. The main concern of steganography is the undetectability of the hidden content [13]. Many steganography methods act on the spatial domain of the image [14–16]. In most of these methods, the hidden content is embedded by changing the least significant bits of pixel values. Embedding at the spatial level is sufficient to deceive the HVS. However, the resistance of these methods to attacks are weak. More advanced methods in steganography uses the spatial frequency domain where embedding is performed at the cosine transform level [17–19]. McKeon [20] shows how to use the discrete Fourier transform for steganography in movies. Some adaptive steganography methods consider the statistics of image

features before embedding the message [21, 22]. For example, the noisy parts of an image are more suitable for embedding than the smooth parts [23].

Digital watermarking. Digital watermarking is used for the protection of digital content. Different than steganography, the visible content is more important than the hidden content. The strength of watermarking methods is related to the difficulty of removing hidden content from the visible content. The watermark aims at marking the digital content with an ownership tag. Copyright protection, fingerprinting to trace the source of illegal copies, and broadcast monitoring are the main purposes of digital watermarking [24]. In reversible watermarking techniques, a complete restoration of the visible content is possible with the extraction of the watermark [25]. Several approaches use lossless image compression techniques to create space for the watermarking data [26–28]. Other work is based on histogram modification [16, 29, 30] and quantization algorithms [25] to embed watermark data. Although many different algorithms are used, the main goal of all reversible watermarking methods is the same: avoiding damaging sensitive information in the visible content and enabling a full extraction of the watermark and original data. For the extraction of the watermark, a retrieval function is required. Complex embedding functions result in complex retrieval functions requiring special software. This is one of the disadvantages of digital watermarking techniques. Although they provide a high level of security, the originality cannot be controlled as easily as in many security printing techniques. In many security printing techniques, the originality of the content can be verified by humans with simple tools such as a magnifier, a directed light, or a UV light.

Our method is neither competing with watermarking nor with conventional steganography. Visual watermarking protects the content of an original video. We however generate synthetic signals to hide visual information. Most stenographic methods use very complex decoding procedures. But, our method aims at revealing information without using any decoding algorithm, i.e. by long exposure photography. Therefore, our method cannot be replaced with existing visual watermarking or steganography methods.

Visual illusions Our work is also related to the domain of optical visual illusions. Visual illusions have received much attention from artists and researchers working in cognitive science, neuroscience, and computer science. Visual illusions can be created by exploiting variations of the visual response according to the viewing conditions. For example, Oliva et al. [31] take advantage of the frequency dependent visual sensitivity of the HVS. They choose two different frequency bands to represent two different images that are combined. At one distance, one image and at the other distance, the other image appears. Similarly, feature and conjunction search are two different mechanisms of the HVS to recognize objects [32]. The feature search mechanism is fast and performs parallel processing by checking single features, while the conjunction search mechanism is slow and interprets multiple features to find objects. Chu et al. [33] rely on these mechanisms to embed and hide objects in other percepts. By removing several features allowing fast recognition, a hidden image remains imperceptible for a certain time. In the work by Mitra et al. [34], the hidden image is only perceieved when looking onto the image as a whole. Arpa et al. [35] use the difference between night vision and day vision of the HVS. They conceive a combined image that is perceived as one image at night and as a different image during the day.

3. SELF MASKING

We now describe our approach for hiding an image in a video. The hidden information is not perceivable by the human eye but the pixelwise average of the video over some time interval reveals the hidden image. With the correct exposure time, conventional and digital cameras can detect the hidden information. Software averaging over the video frames also reveals the image.

The main challenge resides in masking the input image by spatio-temporal signals that are a function of the input image. To achieve this, we present a visual masking process that enables hiding the input image for both spatial and temporal perception.

In conventional visual masking methods, masks and targets are different stimuli. However, in our method, the mask is defined as a function of the target image. We call this approach *self masking*.

We initially define the problem in the continuous domain. A constant target signal p is reproduced by the integration of $f(t)$, a time dependent masking function, over a duration τ:

$$p = \frac{1}{\tau}\int_0^\tau f(t+\delta)dt \tag{1}$$

In order to create spatial noise, a phase shift parameter δ is selected randomly at each spatial position. We assume that the display is linear. The target signal p, the duration τ, and the phase shift δ are known parameters. The challenge resides in finding a function $f(t+\delta)$ satisfying this integration and ensuring that the target signal is masked at each time. In section 5, we discuss the different alternatives for the masking function $f(t+\delta)$.

In practice, our signals are not continuous since the target is a digital image and the mask is a digital video designed for modern displays. Let I be a target image to be masked into a video V having n frames. Initially, we reduce the contrast of the input image I by linear scaling. As discussed in Section 4, this is required in order to reach the *masking threshold*.

A multi-band masking approach is required to mask both high frequency and low frequency contents. Applying the masking function solely on input pixels would only mask high frequency content. Therefore, we decompose the low contrast target image I_c into spatial frequency bands. A Gaussian pyramid is computed from the target image I_c. To obtain the frequency bands, we compute the differences of every two neighbouring pyramid levels. In practice, we use a standart Laplacian pyramid with a 1-octave spacing between frequency bands [36]. Finally, for each contrast reduced pixel value $I_c^l(x,y)$ in each band l, we solve a discretized instance of Eq. (1). Let $t_1,...,t_n$ be a set of n uniformly spaced time points. Then the integral in Eq. (1) is approximated as follows

$$I_c^l(x,y) = \frac{1}{n}\sum_{i=1}^{n} v_i^l(x,y) \tag{2}$$

$$v_i^l(x,y) = f(t_i + \delta_l(x,y)) \tag{3}$$

where v_i^l is a frequency band l of the frame v_i at time point t_i of the resulting video. A different phase shift value δ is assigned to each pixel (x,y) in each band l.

Once all bands v_i^l of each frame v_i are constructed, we sum the corresponding bands to obtain the final frame at time point t_i:

$$v_i = \sum_{l=1}^{k} v_i^l \tag{4}$$

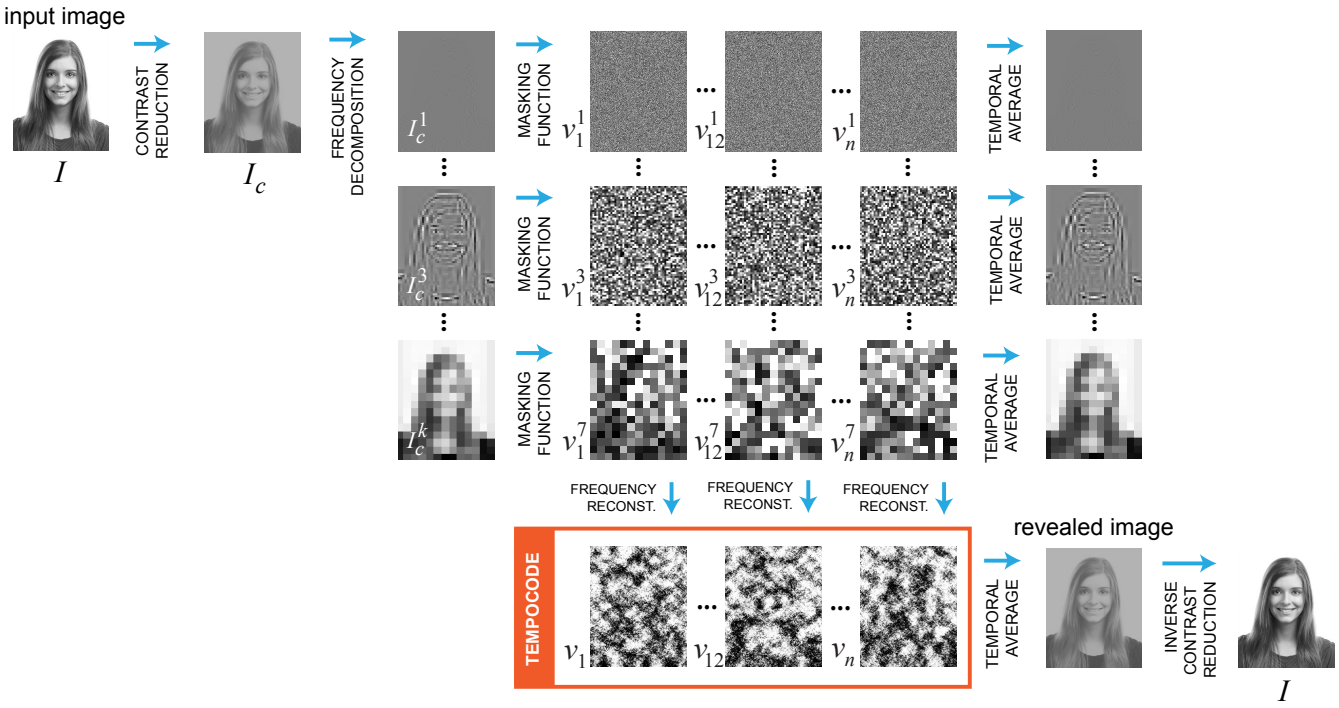


Fig. 4. An overview of our model. A tempocode is generated for an input image I. The resulting video has $n = 24$ frames and is constructed with $k = 7$ frequency bands. In the figure, only 3 frames and 3 frequency bands are shown.

where k is the number of bands. An overview of the model is given in Fig. 4.

4. CONTRAST REDUCTION FOR MASKING PURPOSES

As presented in Section 2, a masking signal with a certain contrast can mask a target signal having a contrast smaller than the *masking threshold*. In our model, we always generate our mask with 100 percent contrast in order to enable a maximum contrast of the target image. We first reduce the contrast of the target image I and move the contrast reduced image to the center of the available range:

$$I_c(x,y) = \alpha \cdot I(x,y) + \frac{1}{2} - \frac{\alpha}{2} \tag{5}$$

where α is the reduction factor and $0 < \alpha < 1$.

The amount of contrast reduction α depends on the contrast, spatial frequency, and orientation of the target (see Sec. 2).

It is very important to select the correct contrast reduction factor α to reach the *masking threshold*. However, the input image consists of a mixture of locally varying contrasts, spatial frequencies, and orientations that affect masking. The contrast reduction factor α should be selected by considering the local image element that requires the largest amount of the contrast reduction. Once this image element is masked, all other image elements are masked as well.

Since our mask always has the maximum contrast, its contrast is not a variable in our model.

5. MASKING FUNCTIONS

Many different types of masking functions $f(t+\delta)$ fulfill the requirements of Eq. (1). We can define a random function with uniform probability, a Gaussian function, a Bezier curve, a logarithmic function, or periodic functions such as a square wave, a triangle wave, or a sine wave. However, the following constraints need to be satisfied:

- Eq. (1) must have a solution for the selected function within the dynamic range of each frequency band.
- Masking must be achieved during the whole video V. In other words, any visual element that could reveal the target image I must remain invisible to the human eye.

In the following, we describe random, periodic, and dither masking functions.

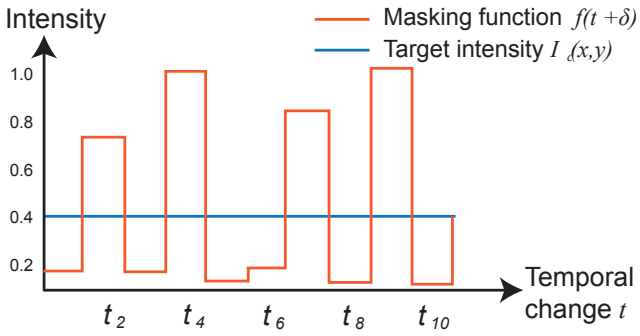


Fig. 5. Example of a discontinuous random function $f(t)$ to mask the target image.

A. Random masking function

Our random masking function is made of n random uniformly distributed samples varying temporally for each pixel of each

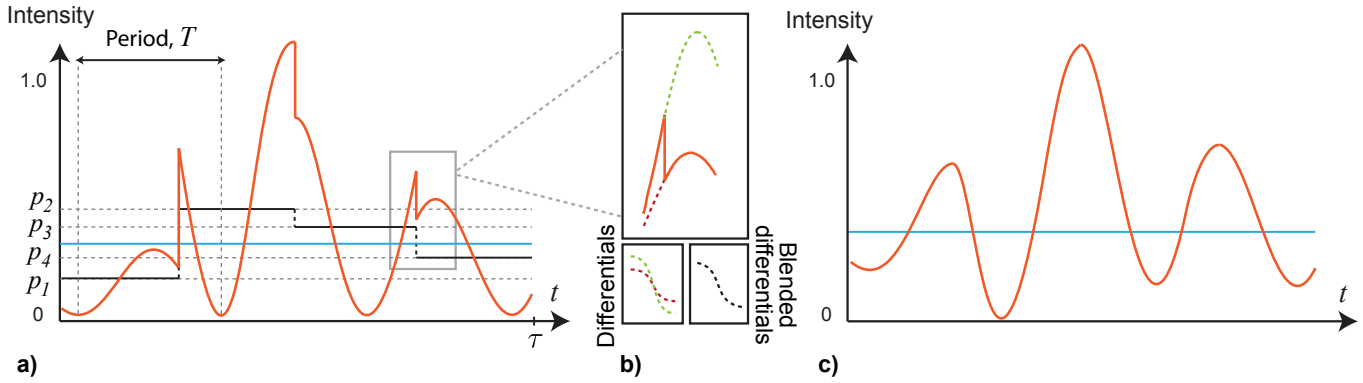


Fig. 6. The integration of a modulated wave yields the given target intensity $I_c^l(x,y)$ (blue line). a) First 4 parent-samples p_1, p_2, p_3, p_4 are generated. Their average gives the target intensity. Then for each parent sample, a simple sinusoid is generated by ensuring that its integration yields the parent sample. b) A refinement process over the modulated wave is applied to remove the discontinuities. c) After the optimization process, we obtain a smooth modulated wave.

band. The mean of this uniform distribution is given by the intensity $I_c^l(x,y)$ of the corresponding pixel of band l. Eq. (2) holds with an error that depends on the number of samples. If the number of samples is small, the error becomes larger. To enforce Eq. (2), we redistribute the error over all samples. Besides, the samples whose values are out of the allowed range are clipped. The remainders are redistributed equally to the other samples. This process is repeated until all samples are within the allowed range.

As shown in Fig. 8b, if the contrast of the target image is sufficiently reduced, the random function masks to a large extent the target image. However, this is only true when each frame is observed separately. When all frames are played as a video (e.g., at $30fps$), the target image is slightly revealed. This is due to the fact that the target image is well masked spatially but not temporally. The human visual system has a temporal integration interval of $40 \pm 10ms$ [11]. Therefore a few consecutive frames can be averaged by the HVS. If we look at the signal in Fig. 5, the average of any two consecutive frames has a value close to the target intensity. A low frequency masking function is therefore required to ensure temporal masking.

B. A sinusoidal composite wave

As we have seen in the previous section, a temporally continuous low frequency masking signal is required to avoid revealing the target signal by temporal integration of the HVS. We thus propose a periodic function that results in spatial discontinuity and temporal continuity of the resulting video.

We use a sine function as our periodic function. Spatial juxtaposition of phase-shifted sine functions may reveal local parts of the target image. Therefore, instead of using a regular sine function, we create a sinusoidal composite wave by varying the function in amplitude for a given number of segments.

In order to create m sine segments varying in amplitude, we first generate m uniformly distributed random parent-samples p_j for each pixel of each band ensuring that their mean is $I_c^l(x,y)$:

$$I_c^l(x,y) = \frac{1}{m}\sum_{j=1}^{m} p_j \tag{6}$$

Since we have a small number of parent-samples (e.g., 4), the mean $I_c^l(x,y)$ will not hold. Therefore, we redistribute the error across the samples. Next, for each parent-sample p_j, we establish a function $f_j(t+\delta)$ in the form of Eq. (1) such that:

$$p_j = \frac{1}{\tau_e^j - \tau_s^j}\int_{\tau_s^j}^{\tau_e^j} f_j(t+\delta)dt \tag{7}$$

where $\tau_s^j = (j-1) \times \frac{\tau}{m}$ is the start time, $\tau_e^j = (j) \times \frac{\tau}{m}$ is the end time, $j \in [1,..,m]$ is the index of each parent-sample, and τ is the total duration of the video to be averaged.

We define the masking function $f_j(t+\delta)$ for each parent sample as a continuous section of a sine in a form that is analytically integrable and lies within the dynamic range for most of its values.

$$f_j(t+\delta) = k_j \cdot sin(2\pi\frac{t}{T} + \delta) + k_j \tag{8}$$

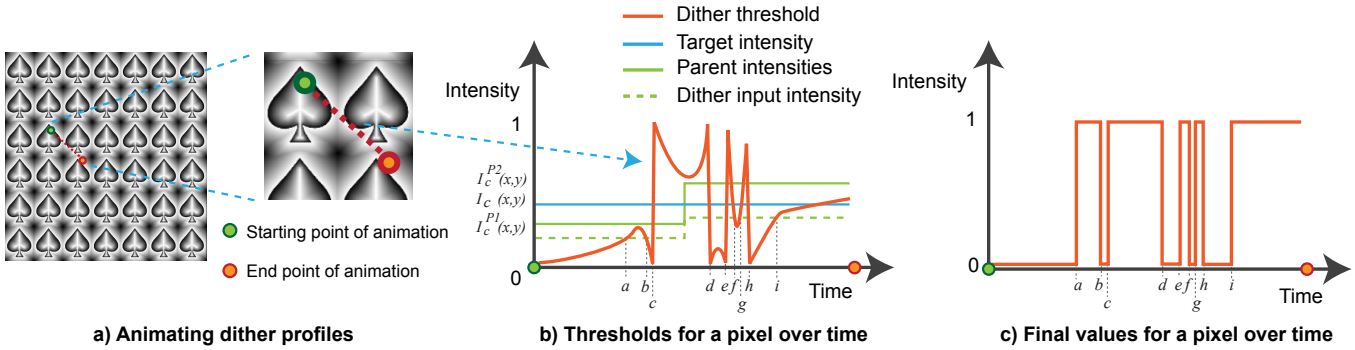
where k_j is the amplitude and T is the period. As shown in Fig 6a, the period T and the duration of video τ have different values. The total duration τ of video is given by the user.

By inserting Eq. (8) into Eq. (7), we can express k_j in function of the other parameters:

$$k_j = \frac{p_j(\tau_s^j - \tau_e^j)}{\tau_s^j - \tau_e^j + \frac{T(\cos(\frac{2\pi\tau_e^j}{T}+\delta) - \cos(\frac{2\pi\tau_s^j}{T}+\delta))}{2\pi}} \tag{9}$$

For each pixel of each frequency band, these m functions $f_j(t+\delta)$ of parent samples are sampled with one sample per video frame (Fig 6a). The averages are enforced by redistributing the errors over the samples. According to Eq. (7), the average of each sinusoidal section gives the value of a parent sample. Thus, the average of all n samples gives the target intensity of the considered band $I_c^l(x,y)$.

In order to ensure a phase continuity between the sinusoidal segments, we select the phase shift δ randomly only for the first sinusoidal segment $f_j(t+\delta)$. For all other functions associated to parent samples we use the current phase δ and the current period T. Nevertheless, due to the variations of the amplitudes, we obtain a non-continuous composite signal. These discontinuities appear at the junctions between successive sinusoidal segments (see Fig 6a) and are visible in the final output video.



a) Animating dither profiles b) Thresholds for a pixel over time c) Final values for a pixel over time

Fig. 7. a) An artistic dither matrix representing a spade is repeated horizontally and vertically to cover the whole frame. b) For each frame of the output video, the array of dither matrices is translated in the direction of the desired animation. This creates for each pixel a succession of dither thresholds over time. An optimum dither input intensity is determined for the two parent samples separately. c) The pixels above dither input intensity are assigned black and the others white. The overall average gives the target intensity.

To remove the discontinuities at the junction points, we apply a refinement process by using differential values. From the samples of the composite wave, we first calculate the differential values by taking the backward differences: $\Delta v_i^l(x,y) = v_i^l(x,y) - v_{i-1}^l(x,y)$. We then blend the differential values of the end part of a sinusoidal segment with those at the starting part of the following sinusoidal segment (see Fig. 6b).

With the blended differential values, we re-calculate the intensity values for each pixel of each band by minimizing the following optimization function:

$$\begin{aligned} E(v_1^l,...,v_n^l) = &\sum_{i=1}^{n} \|\Delta v_i^l(x,y)' - \Delta v_i^l(x,y)\|^2 \\ &+ \|I_c^l(x,y) - \frac{1}{n}\sum_{i=1}^{n} v_i^l(x,y)'\|^2 \\ &+ \sum_{b=1}^{m} \|v_b^l(x,y)' - v_b^l(x,y)\|^2 \end{aligned} \quad \textbf{(10)}$$

where n is the total number of frames. The first term in the optimization minimizes the square differences between blended differential values $\Delta v_i^l(x,y)$ and the differential values $\Delta v^l(x,y)_i'$ of the new intensities in the solution set. The second term is a constraint to guarantee that the overall average $I_c^l(x,y)$ of the new intensities $v^l(x,y)_i'$ is still satisfied. The third term preserves the overall shape of the signal, by fixing the center sample of each sinusoidal segment as a constraint. b represents the index of center sample for each parent sample.

This optimization is solved as a sparse linear system. We obtain a smooth signal (Fig. 6c). The deviations from the average $I_c^l(x,y)$ caused by the optimization are redistributed over the samples.

As shown in Fig. 8c (inset), the sinusoidal composite wave successfully masks the target image in both the spatial and temporal domains.

C. Temporal dither masking function

A sinusoidal composite wave enables masking the target image both spatially and temporally. However, the visible part, the tempocode video, does not convey any visual meaning. We thus propose to replace the spatial noise with meaningful patterns. For this purpose, we make use of artistic dither matrices [37].

When printing with bilevel pixels, dithering is used to increase the number of apparent intensities or colors [38]. A full tone color image can be created with spatially distributed surface coverages of cyan, magenta, yellow, and black inks. The HVS integrates the tiny inked and non-inked areas into the desired color.

A dither matrix includes threshold values indicating at which intensity levels pixels should be inked. Artistic dithering enables ordering these threshold levels so that for most levels turned-on pixels depict a meaningful shape [37]. We adapt artistic dithering to provide visual meaning to tempocode videos.

We repeat the selected dither matrix horizontally and vertically to cover the whole frame. We then animate the dither matrices [39]. The animation can be achieved by a uniform displacement of the dither matrices at successive frames. As shown in Fig. 7, for a single pixel, the threshold values vary over time. At any time point of the video, the current dither threshold determines if the pixel is white or black. Accounting for the varying thresholds over time, we can determine a dither input intensity ensuring that the average of the resulting black and white pixels yields the target intensity (Eq. (1)). Instead of finding such a dither input intensity, we directly assign white or black to the successive temporal dither threshold levels as follows:

- Find the ratio r_{wb} of white to black pixels to obtain the target intensity $I_c(x,y)$. Then derive the number w of white pixels. This is calculated as follows:

$$r_{wb} = \frac{I_c(x,y)}{1 - I_c(x,y)} = \frac{w}{n-w} \qquad 0 \le I_c(x,y) \le 1 \quad \textbf{(11)}$$

  where n is the total number of frames.

- For each spatial pixel, sort its dither threshold values that are changing temporally.

- Assign the first w temporal intensity values to white and the rest to black.

- Revert the temporal intensity values back to their original indices (e.g. frame number).

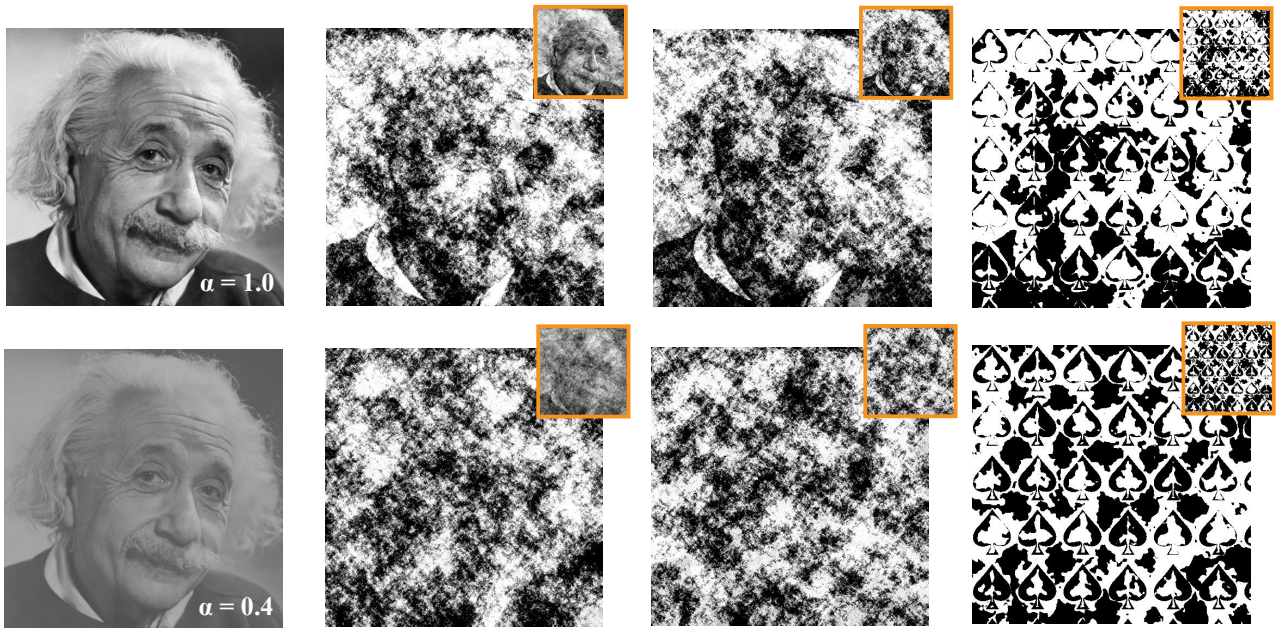


Fig. 8. Sample frames from the tempocodes generated with different masking functions with the following parameters: Duration $\tau = 4s$, frame rate = $60fps$, the period $T = 1.65s$, and the number of frequency bands $k = 7$. In the first row, the results are generated with a target image having no contrast reduction ($\alpha = 1.0$). None of the functions can fully mask the target image. In the second row, the contrast of the target image is reduced ($\alpha = 0.4$). For a single frame, all functions can mask the target image. However, when a few consecutive frames are averaged by the HVS temporal integration, the random function (b) reveals the target image. The two other methods (c and d) are able to hide the target image not just spatially but also temporally. The insets on the right top corner of the frames show the average of 4 consecutive frames as a simulation of the HVS temporal integration.

For an exact 8-bit representation of the target image, $n = 256$ frames are required. With a lower number of frames, one may use additional gray levels or have an approximation of the target image.

This algorithm satisfies one of our conditions, i.e., the average of the frames yield the target image (Eq. (2)). However, a multi-band decomposition cannot be carried out with the dithered binary images since they are bilevel. As shown previously, the multi-band decomposition was an important component for masking the target image. To overcome this problem, we create two parent frames I_c^{P1} and I_c^{P2} from the input image I_c by using the random masking function applied on each band of image I_c, as described in Sec. A. For these two parent frames, due to the multi-band decomposition, the target is masked spatially (e.g., Fig. 8b). Then for each of the two parent frames, we create $\frac{n}{2}$ frames by using the temporal dither function as described above. The levels and corresponding pixel values are shown in Fig 7 for a single pixel. Finally we get n dithered frames forming our final video V in which the target is successfully masked, as shown in Fig. 8d.

6. AVERAGING

Tempocodes can be revealed by averaging. A pixelwise average of the original tempocode video yields the input target image (Fig. 4). For the random function and the sinusoidal composite wave, with floating point numbers, contrast reduction of the input image and masking is performed without losing precision. For dithering, some details of the original image can get lost due to the binarization of the signal (e.g, $n < 256$) and possibly due to low-pass filtering of the target image in order to obtain smooth tempocodes.

Instead of averaging the tempocode by software, a camera can be used to average with long exposure photography and to reveal the target image. The duration of temporal averaging can be controlled with the shutter speed [40]. Furthermore, smartphones can be used for averaging. By placing markers around the video, the projective distortion is removed and the rectified frames are averaged (see the Visualization I).

7. RESULTS

The methods for generating tempocodes are described for grayscale target images. For color images, we use exactly the same procedure, and apply it to each color channel separately. Reaching masking thresholds by reducing the intensity range of the target image is critical to achieve masking. Thus, for color target images, we reduce the intensity range by considering the luminance channel. The chromatic contrast sensitivity of the HVS is lower than the achromatic contrast sensitivity. Once the masking threshold is reached for the luminance channel, the thresholds are also reached for the chromatic channels.

In order to illustrate the dissimilarity between the original image and the computed tempocode results, we use an image similarity metric. We calculate the multi-scale structural similarity index (MS-SSIM) defined by Wang et. al. [41] between the original image and each frame of the output. The MS-SSIM is calculated considering luminance differences, contrast differences,

Fig. 9. Sample tempocode frames (1) generated with the different input images and the different dither matrices. The hidden images can be revealed by averaging (2). An inverse contrast reduction operation yields the original input image (3). In all the cases, the target image is recovered by software averaging the original tempocode frames. We have the following parameters (a) for the woman, $\alpha = 0.4$, (b) for the lion $\alpha = 0.5$, (c) for the QR code $\alpha = 0.2$, and (d) for the text, $\alpha = 0.3$.

structural differences, and viewing conditions. The MS-SSIM yields a similarity index value between 0 and 1. A high MS-SSIM index value means that the given two images are very similar [41]. We evaluate the dissimilarity between the hidden target image and the tempocode frames. We expect MS-SSIM values to be very small when the contrast reduction α is sufficient. We compare each frame of the tempocode with the input image and calculate a mean MS-SSIM index for the whole video by averaging the results of each frame. We define dissimilarity as 1 minus the mean MS-SSIM index. In Fig. 10, the dissimilarity between tempocode frames relying on the sinusoidal composite wave function and the hidden image is shown for three different target images.

For photographic images (i.e., Einstein and lion) the dissimilarity is very high even for a low contrast reduction. The reason is that the frequency bands within the image do not use the full energy while our mask covers the full energy range. The frequency bands are therefore easier to mask. However, for a QR code, this is not the case. Most frequency bands use the full energy range. This requires the contrast of the input image to be strongly reduced (i.e., $\alpha = 0.2$) in order to achieve masking.

Any target image can be masked by reducing its contrast. We let the user select α heuristically. From our experience, $\alpha = 0.5 \pm 0.1$ enables masking most types of photographic images. For images including texts and structures with full contrast

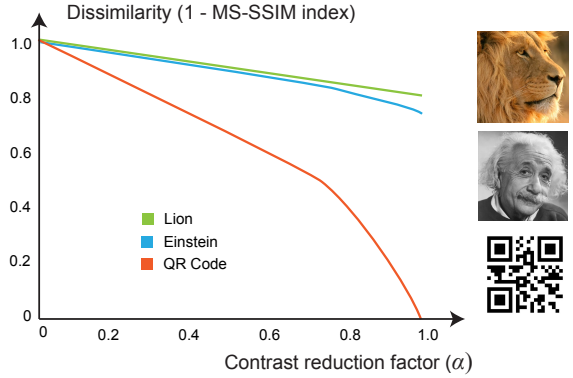


Fig. 10. The dissimilarity expressed as one minus the mean SSIM index for three different target images. Each frame of the output is compared with the target image. We compute the average dissimilarity over all frames. As we decrease α, the dissimilarity increases differently for the target images.

(e.g., binary images), contrast reduction needs to be as much as $\alpha = 0.3 \pm 0.1$. Fig. 9 shows several tempocodes produced with different input images requiring different contrast reduc-

tion factors α. More results and comparisons are given in the Visualization I and Dataset I.

Since our contrast reduction is linear, the original target image can be obtained by multiplying the revealed images by the inverse of the intensity range reduction (i.e., the inverse of Eq. (5)).

Our method is an initial work that introduces a screen-camera channel for hiding information by simple averaging. This can inspire future works for different applications. The encoding is complex, but the decoding is very simple. Thus, hidden images can be revealed by non-expert users, but not created. The present method does not compete with existing watermarking or stenographic methods that require complex decoding procedures. It can be rather used as a first level secure communication feature. More and more security applications such as banking software use smartphones to identify codes that appear on a display. In the present case, instead of directly acquiring the image of a code, the smartphone might acquire a video that incorporates that code. For example, instead of showing a QR code on an electronic document directly, our method can be used to hide it. Hiding a message into a video can be seen as one building block within a larger security framework.

8. CONCLUSION

We propose to hide grayscale or color images in videos whose frames mask them both spatially and temporally. The hidden image can be revealed by temporal pixel-wise averaging of the video. Many companies would like to provide to their customers electronic tickets and identity cards. This necessitates new security features to protect digital documents against forgery. Tempocodes have the potential to become a building block within a document security framework. For authentication, smart phones may reveal the images or messages hidden into the tempocodes.

REFERENCES

1. D. M. Chandler, "Seven challenges in image quality assessment: past, present, and future research," ISRN Signal Processing (2013).
2. P. J. Burt and E. H. Adelson, "The laplacian pyramid as a compact image code," IEEE Transactions on Communications **31**, 532–540 (1983).
3. F. W. Campbell and J. Robson, "Application of fourier analysis to the visibility of gratings," The Journal of Physiology **197**, 551–566 (1968).
4. F. Campbell and J. Kulikowski, "Orientational selectivity of the human visual system," The Journal of Physiology **187**, 437 (1966).
5. R. A. Smith and D. J. Swift, "Spatial-frequency masking and birdsall's theorem," Journal of the Optical Society of America A **2**, 1593–1599 (1985).
6. S. J. Daly, "Visible differences predictor: an algorithm for the assessment of image fidelity," Proc. SPIE/IS&T Symposium on Electronic Imaging: Science and Technology **1666**, 2–15 (1992).
7. T. O. Aydın, "Human visual system models in computer graphics," Ph.D. thesis, Max Planck Institute for Computer Science (2010).
8. D. G. Pelli and B. Farell, "Why use noise?" Journal of the Optical Society of America A **16**, 647–653 (1999).
9. A. B. Watson, R. Borthwick, and M. Taylor, "Image quality and entropy masking," Proc. SPIE Human Vision and Electronic Imaging II **3016**, 2–12 (1997).
10. A. B. Watson and J. A. Solomon, "Model of visual contrast gain control and pattern masking," Journal of the Optical Society of America A **14**, 2379–2391 (1997).
11. A. Gorea and C. W. Tyler, "New look at bloch's law for contrast," Journal of the Optical Society of America A **3**, 52–61 (1986).
12. M. Kalloniatis and C. Luu, "Temporal resolution," in "Webvision," H. Kolb, E. Fernandez, and R. Nelson, eds. (University of Utah Health Sciences Center, 1995).
13. A. Cheddad, J. Condell, K. Curran, and P. Mc Kevitt, "Digital image steganography: Survey and analysis of current methods," Signal Processing **90**, 727–752 (2010).
14. J. Fridrich, M. Goljan, and D. Hogea, "Steganalysis of jpeg images: Breaking the f5 algorithm," Proc. Information Hiding pp. 310–323 (2003).
15. Z. Li, X. Chen, X. Pan, and X. Zeng, "Lossless data hiding scheme based on adjacent pixel difference," Proc. International Conference on Computer Engineering and Technology **1**, 588–592 (2009).
16. P. Tsai, Y.-C. Hu, and H.-L. Yeh, "Reversible image hiding scheme using predictive coding and histogram shifting," Signal Processing **89**, 1129–1143 (2009).
17. X. Li and J. Wang, "A steganographic method based upon jpeg and particle swarm optimization algorithm," Information Sciences **177**, 3099–3109 (2007).
18. A. Hashad, A. S. Madani, A. E. M. A. Wahdan *et al.*, "A robust steganography technique using discrete cosine transform insertion," Proc. IEEE International Conference on Information and Communications Technology pp. 255–264 (2005).
19. K. Raja, C. Chowdary, K. Venugopal, and L. Patnaik, "A secure image steganography using lsb, dct and compression techniques on raw images," Proc. IEEE International Conference on Intelligent Sensing and Information Processing pp. 170–176 (2005).
20. R. T. McKeon, "Strange fourier steganography in movies," Proc. IEEE International Conference on Electro/Information Technology pp. 178–182 (2007).
21. P. Sallee, "Model-based steganography," in "Digital Watermarking," (Springer, 2004), pp. 154–167.
22. H. Hirohisa, "A data embedding method using bpcs principle with new complexity measures," Proc. Pacific Rim Workshop on Digital Steganography pp. 30–47 (2002).
23. P. Wayner, *Disappearing cryptography: information hiding: steganography & watermarking* (Morgan Kaufmann, 2009).
24. G. C. Langelaar, I. Setyawan, and R. L. Lagendijk, "Watermarking digital image and video data. a state-of-the-art overview," IEEE Signal Processing Magazine, 17 (5) (2000).
25. A. Khan, A. Siddiqa, S. Munib, and S. A. Malik, "A recent survey of reversible watermarking techniques," Information Sciences **279**, 251–272 (2014).
26. M. Arsalan, S. A. Malik, and A. Khan, "Intelligent reversible watermarking in integer wavelet domain for medical images," Journal of Systems and Software **85**, 883–894 (2012).
27. M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, "Lossless generalized-lsb data embedding," Image Processing, IEEE Transactions on **14**, 253–266 (2005).
28. G. Xuan, C. Yang, Y. Zhen, Y. Q. Shi, and Z. Ni, "Reversible data hiding using integer wavelet transform and companding technique," Proc. Digital Watermarking pp. 115–124 (2005).
29. C.-C. Lin, W.-L. Tai, and C.-C. Chang, "Multilevel reversible data hiding based on histogram modification of difference images," Pattern Recognition **41**, 3582–3591 (2008).
30. A. Khan, S. A. Malik *et al.*, "A high capacity reversible watermarking approach for authenticating images: Exploiting down-sampling, histogram processing, and block selection," Information Sciences **256**, 162–183 (2014).
31. A. Oliva, A. Torralba, and P. G. Schyns, "Hybrid images," ACM Transactions on Graphics **25**, 527–532 (2006).
32. A. M. Treisman and G. Gelade, "A feature-integration theory of attention," Cognitive Psychology **12**, 97–136 (1980).
33. H.-K. Chu, W.-H. Hsu, N. J. Mitra, D. Cohen-Or, T.-T. Wong, and T.-Y. Lee, "Camouflage images," ACM Transactions on Graphics **29**, 51 (2010).
34. N. J. Mitra, H.-K. Chu, T.-Y. Lee, L. Wolf, H. Yeshurun, and D. Cohen-Or, "Emerging images," ACM Transactions on Graphics (TOG) **28**, 163 (2009).
35. S. Arpa, T. Ritschel, K. Myszkowski, T. Çapın, and H.-P. Seidel, "Purkinje images: Conveying different content for different luminance adaptations in a single image," Computer Graphics Forum **34**, 116–126 (2015).
36. M. N. Do and M. Vetterli, "Framing pyramids," IEEE Transactions on

Signal Processing **51**, 2329–2342 (2003).
37. V. Ostromoukhov and R. D. Hersch, "Multi-color and artistic dithering," Proc. 26th Annual Conference on Computer Graphics and Interactive Techniques pp. 425–432 (1999).
38. C. Hains, S. Wang, and K. Knox, *Digital Color Halftones, Chapter 6 in Digital Color Imaging Handbook* (Boca Raton, FL: CRC, 2003).
39. R. D. Hersch and B. Wittwer, "Method and computing system for creating and displaying images with animated microstructures," (2009). US Patent 7,623,739.
40. E. Allen and S. Triantaphillidou, *The Manual of Photography and Digital Imaging* (CRC Press, 2012).
41. Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," Proc. Signals, Thirty-Seventh Asilomar Conference on Systems and Computers **2**, 1398–1402 (2003).