Hiding Data in Images Using Spectral Filtering and Deep Neural Networks

Hadar Shalev
Dept. of Computer Science
University of Haifa
Haifa, Israel
hshadar@gmail.com

Pe'erly Setter
Dept. of Psychology
University of Haifa
Haifa, Israel
psetter@univ.haifa.ac.il

Ruth Kimchi
Dept. of Psychology
University of Haifa
Haifa, Israel
rkimchi@univ.haifa.ac.il

Hagit Hel-Or
Dept. of Computer Science
University of Haifa
Haifa, Israel
hagit@cs.haifa.ac.il

***Abstract*—Privacy preserving and data security is one of the major concerns of our world. In this study we deal with one aspect, namely *shoulder surfing*, where sensitive information may be accessed by looking at the display or keyboard of the user. We propose a method of hiding information in color images so that it can not be perceived by the naked eye and requires a spectral filter to be seen. We term such images *Spectral Hiding Images*. In this work we developed a system which can automatically generate Spectral Hiding Images. We focus on a basic class of images containing a single numeric digit. We train three deep networks to determine the salient digit in an image, determine the hidden and masked digits in a spectral hiding image and to generate diverse spectral hiding digit images. Mass producing such Spectral Hiding Images using our system will allow for screen content hiding and password hiding. Additionally we show that several Gestalt principles of human perception, are expressed in the trained networks' behavior.**

***Index Terms*—Data hiding, color images, spectral filtering, shoulder surfing, user privacy, deep neural networks.**

I. Introduction

A main concern in privacy protection is known as "shoulder surfing", in which a hostile agent looks over the shoulder of a legitimate user and acquires sensitive information. This may be passwords or any other displayed information (e.g. when using computer displays in public locations). We introduce an approach to data hiding which may be used to prevent shoulder surfing. It is user friendly and requires no complex technology. The approach is based on exploiting the color spectra of the display, together with simple color filter glasses that can be worn by the users.

The notion of hiding information using spectral filtering is not new. However, in most previous methods, the hidden data can be seen without any filter or image quality, when viewed with a filter, is poor. Additionally in most existing solutions, the image is manually created therefore cannot be mass produced. The novelty of our approach is in using deep learning architectures. This allows the following advantages:

- The hidden data can not be seen without the filter, even with prior knowledge of the hidden data.
- The hidden data, when viewed through the filter, is clearly seen.
- Creation of these spectrally hidden images is performed automatically so that they can be mass produced.

To achieve our goal we train several deep neural networks on data labeled by human subjects and on manually created data. The networks perform the following tasks:

- Recognize the visible data and the spectrally hidden data in a specific image.
- Create new synthetic images in which visual data is spectrally hidden from the illegitimate user.

Finally, we show that the trained neural networks have learned principles from the laws of human visual perception, namely Gestalt principles.

This research was supported by grant no 1455/16 from the Israeli Science Foundation.

II. Background

A. Spectral Filtering

The Spectral Power Distribution (SPD) of a light signal is a function $I(\lambda)$ that defines the energy at each spectral wavelength (Figure 1a). A spectral filter attenuates the power at every spectral wavelength and is represented by a function $f(\lambda)$ that defines the attenuation factor for each spectral wavelength (Figure 1b). The outcome of applying spectral filtering with filter $f(\lambda)$ on input SPD $I(\lambda)$ is given by:

$$I_f(\lambda) = I(\lambda) \cdot f(\lambda)$$

An example is shown in Figure 1.

Although SPDs represent high dimensional data, the space of perceived colors has been shown to be three dimensional [1]. Specifically, it has been shown that almost all perceived colors can be produced as a linear combination of three SPDs, often referred to as Primaries, which can be viewed as the

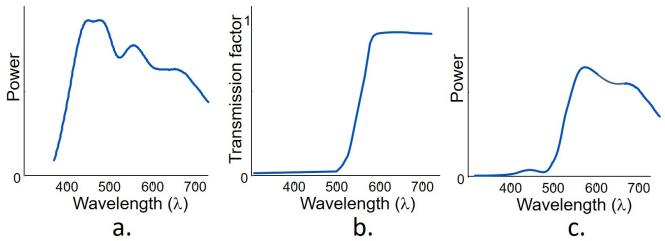

Fig. 1. Spectral Filtering. a. Spectral Power Distribution (SPD) of light signal. b. Filter transmission function. c. Resulting spectrally filtered SPD.

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

Fig. 2. Top: An RGB image viewed without the spectral filter. Bottom: The image viewed through a red spectral filter. The filter attenuates the Green channel and blocks the Blue channel. Pixels with high red content are perceived past the filter as bright, while pixels with low Red content are seen as dark. (Best viewed in color).

basis of a 3D color space. The typical 3D color space is the RGB color space with R,G,B representing the Red, Green and Blue components of the color signal. These spaces have been standardized e.g. the CIE-RGB standard [2]. Thus the color of every point in a scene, an image or any viewed object can be represented as 3 values (e.g R,G,B), representing the intensity of the three primaries composing the color. This property allows to present spectral data in digital technologies (monitors, printers etc.).

In this study we use a "red" filter that strongly attenuates all short and middle spectral wavelengths. Since the filter will be used to view an image displayed on a color monitor, we model the effect of the spectral filter as transmitting all R channel content, and a small fraction of the G channel of the image. The filter completely blocks the B channel of the image. The results in this study are robust under different variants of "red" filter transmission functions. An example of the red filter's effect on a displayed RGB image is shown in Figure 2. We represent the filtered image as a gray scale image.

B. Gestalt Principles

Gestalt principles [3] are a set of laws that govern which elements of an image are perceived as grouped. There are numerous principles of Gestalt, however, in this work we consider only the following four factors: proximity, similarity, closure and continuity (see Figure 3):

1) Proximity - objects which are close to each other will usually be grouped together, whether they have or do not have a defined relationship (e.g. in terms of color, size or texture).
2) Similarity - objects which are visually similar to each other (e.g. in terms of color, size or texture) will usually be grouped together.
3) Closure - objects tend to be grouped together when they are perceived as a whole object or might complete a whole object, even when parts of that object do not exist.
4) Continuity - objects tend to be grouped together if they align and form a continuous line or curve.

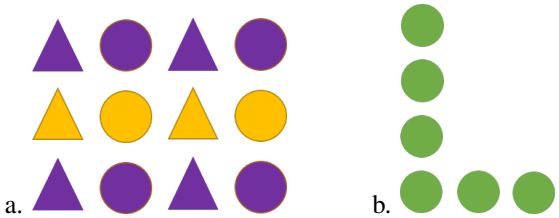

Fig. 4. Gestalt conflict (a) and interaction (b). a) Elements may be grouped into rows, according to color similarity or into columns, according to shape similarity. b) Due to color similarity and proximity, all circles are viewed as a single group.

The strength of grouping under these principles is affected by several factors, including the number of segments, their distance, and their similarity in terms of color, shape or other characteristics. In many cases conflicts may arise between Gestalt principles. An example of a conflict between shape and color similarity is shown in Figure 4. Many studies have investigated Gestalt principles and have attempted to model these principles, their conflicts and interaction [4] [5], but a clear model or set of rules determining how and when these principals interact together is still ongoing research. Furthermore, current studies typically focus on binary (black and white) shapes and often do not take color into consideration. Research in the direction of this paper, may assist in understanding the role of color in Gestalt principles.

III. Related Work

The study presented here is strongly related to Steganography (see [8] for a summary) in which secret data is hidden in some media. Automatic visual hiding of data in images has been previously suggested, based on spatial image content and textures. Several methods such as [9] [10] require highly textured backgrounds, in order to hide the visual data. Other methods [11] create a highly textured image which includes a tailored highly textured background in which the desired

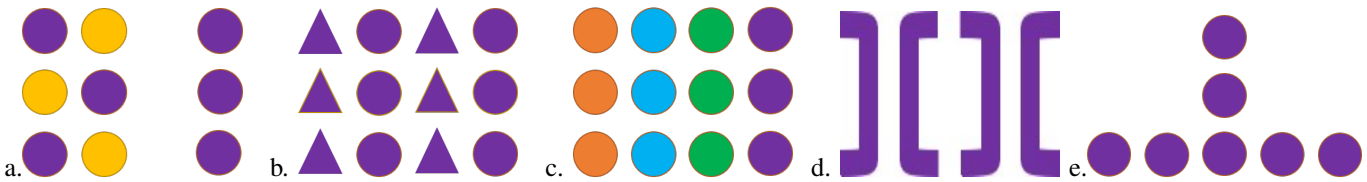

Fig. 3. Gestalt principles. Elements are perceived as grouped into columns due to a) proximity b) shape similarity, and c) color similarity. d) The 2 elements in the middle are grouped due to closure rather than to the closer elements on the sides. e) Elements are grouped into a horizontal and a vertical line due to continuity although the left most element is closer to the top element.

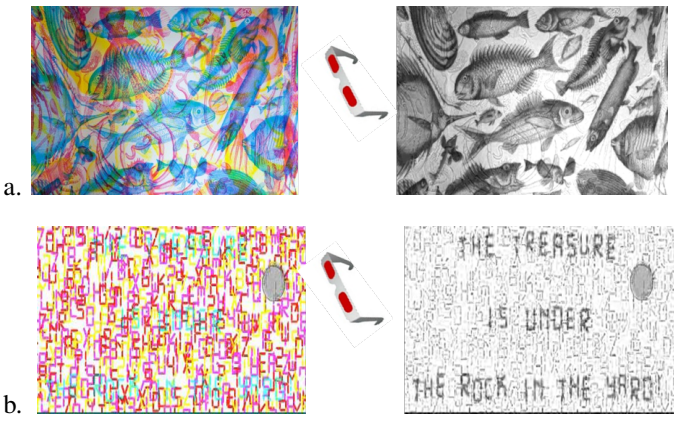

Fig. 5. Previous spectral hiding methods. The original image (left) and the image viewed through the spectral filter (right). Image source [6]. The hidden data can be seen without a filter (top). Image quality after filter is poor (bottom). Image source [7].

object is hidden. These methods were developed to impede automatic detection by a bot, but are not immune to detection by humans. Our proposed method, using spectral hiding, does not use background texture to hide the content. Additionally, when the legitimate user views the image via the filter, the hidden data is clear and can be perceived without effort. The illegitimate user will not be able to see the hidden data, regardless of the time spent observing the image beforehand.

IV. Spectral Data Hiding

In this research we introduce a method of data hiding in which data is "spectrally hidden" in an image and can be viewed only with a spectral filter. An example is shown in Figure 6. The original color figure is shown on the left and the hidden image as seen through the red-pass spectral filter is shown on the right.

We denote by *hidden data*, the data which is visible only with the spectral filter and by *masking data* the data which is visible without the spectral filter. The image is designed such that image elements that compose the masking data distract the viewer's attention from the hidden data in the image. Considering the characteristics of the spectral filter, the relationship between image components creating masking data and hidden data may take several forms:

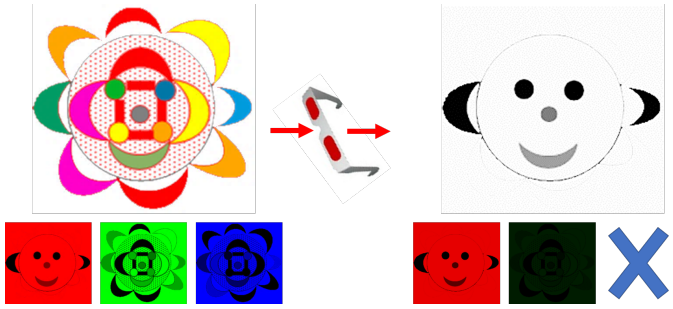

Fig. 6. Spectral hiding image. Original image (left) and image viewed through the spectral filter (right). Corresponding RGB channels are shown below. (Best viewed in color using a "red" filter).

- *Remaining component*: an image component may remain visible in the hidden data, such as the green and cyan components in Figure 6.
- *Disappearing component*: an image component may disappear in the hidden data because it was blocked by the filter. For example, the pink, red, orange and yellow elements in Figure 6.
- *Appearing component*: an image component barely visible in the masking data may appear more pronounced in the hidden data. For example, the gray circle in the middle of Figure 6 and segment 5 in Figure 9. The Appearing component becomes more visible in the hidden data because the filter projects the data from RGB color space to grayscale thus losing some of the color difference between segment and background. In the hidden image, the contrast between the appearing component and the background is similar to that of the remaining components and thus it becomes more visible.

V. Data Representation and Creation

In this research we restricted our data (visible and hidden) to seven-segment digit images. This type of data has a clear structure, therefore, there is a compact representation for each data sample. Instead of a $W \times H \times 3$ representation of a natural color image (where W and H represent width and height respectively), we represent each seven-segment digit using a vector of 24 integers $\in [0, 255]$ representing the RGB values of the 7 digit-segments and the background. In order to represent digits with less than 7 visible segments (all digits except 8), we set the color of the relevant segment to have the same color as the background. Denote by *digit segments* the seven segments that compose the digit numbered 1 to 7, and by *background segment*, the background of the image which is numbered 8. See example in Figure 7.

This representation of the digit-image is memory efficient and therefore advantageous for computation and specifically in training neural networks. However, our proposed method is not limited to seven-segment digits and can be applied to any structured data.

The digit images can be divided into two types (see Figure 8):

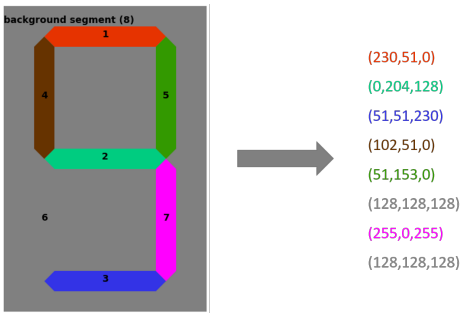

Fig. 7. A digit image is composed of 8 segments: 7 digit segments and a background segment. Each segment is numbered (1 to 8) and is colored (R,G,B). In this example digit segment 6 shares the same color as the background segment and is therefore not visible.

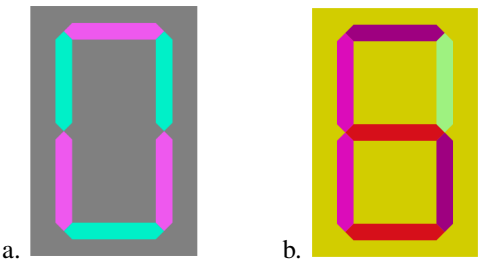

Fig. 8. Simple (a) and ambiguous (b) data samples. Subjects disagree on the most salient digit in the ambiguous data sample: 6 or 8.

- Simple data: the most salient digit in the digit image is consistent across all viewers. Simple data can be colored digit images as well as grayscale digit images.
- Ambiguous data: viewers disagree on the most salient digit in an image.

To determine the most salient digit perceived in each digit-image of our data-set, a user study was conducted (Section VI).

In our research we aim to hide information in an image. Specifically, we wish to create digit images such that the most salient digit in the image is different when viewed with a spectral filter and without. We will term such images as *Spectral Hiding Digit Images*.

Each such digit image is associated with two labels: the *masking digit* perceived when viewing the image with no spectral color filter, and the *hidden digit* which is perceived when looking at the image with a spectral color filter.

In this study we consider images in which the hidden digit image has only one possible salient digit, in contrast to the masking image, which can be associated with more than one salient digit. An example of a spectral hiding digit image is shown in Figure 9. The most salient hidden digit is 9 and masking digit is not 9 (either 5, 6 or 8).

The possible pairings between masking digit and hidden digit is constrained due to the physical characteristics of the spectral filter. Not all pairings are possible (e.g. digit "1" can not mask digit "0"). The possible parings are shown in Table I. The digit pairing can be divided into two types, according to masking principles used to create it:

- Disappear Only (*DO*): in this type, several digit segments that are visible in the masking image "disappear" and are no longer visible in the hidden image. The remaining digit segments are visible both in the masking and hidden images. For example, with the digit pair "3-1", segments number 1,2,3 "disappear" and segments number 5,7 remain. Samples of this type are easily seen with the filter due to high color contrast both in the masking and the hidden images.
  Trivial Disappear Only (*TDO*): pairs of digits are such that there is only one possible digit that can be hidden using Disappear Only segments. Thus the observer may deduce the hidden digit from the masking image. For example, with the pair "7-1", the only digit obtainable from 7 by removing segments is 1.
- Appear and Disappear (*AD*): in this type, several digit segments that were visible in the masking image "disappear" and are no longer visible in the hidden image. Several other digit segments that were almost invisible in the masking image become clearly visible in the hidden image. The remaining digit segments are visible both in the masking and hidden images. Samples of this type are less clearly seen with the filter but are more private due to their compatibility only with a narrow range of color filters.

Due to the characteristics of the spectral filter described in Section IV, it is not possible to create "Appear Only" (**AO**) hiding data samples. Table I shows all possible pairings of masking and hidden digits.

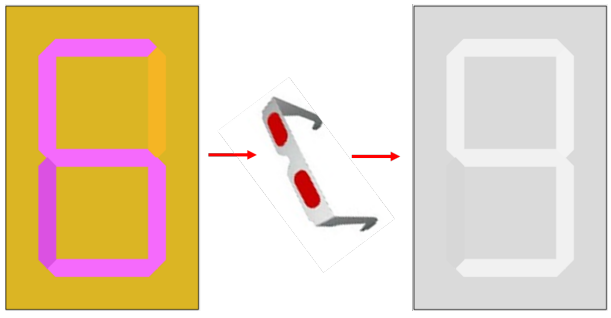

Fig. 9. Spectral hiding digit image example. A digit image viewed without the spectral filter (left) and viewed through the filter (right). The most salient hidden digit is 9 and masking digit is not 9 (either 5, 6 or 8).

VI. Data Collection - User Study

To train the networks used in this research, we required labeled data. Thus we performed a user study in which subjects reported the most salient digit perceived in digit-images. Subjects also had the option of reporting "not-a-digit". Data for the user study was created in Lab color space [12] then mapped to RGB color space and displayed on a calibrated monitor. The Lab color space is a perceptual color space in which Euclidean distance correlates with visually perceptual differences. Thus we could control the perceptual distances in the digit-images used in the user study.

The data-set chosen for the user study was composed of four different types of data. Each data type was chosen to represent ambiguous samples due to different factors. These factors include the following:

	0	1	2	3	4	5	6	7	8	9
0	X	DO	AD	AD	AD	AD	AD	DO	**AO**	AD
1	**AO**	X	AD	**AO**	**AO**	AD	AD	**AO**	**AO**	**AO**
2	AD	AD	X	AD	AD	AD	AD	AD	**AO**	AD
3	AD	DO	AD	X	AD	AD	AD	DO	**AO**	**AO**
4	AD	TDO	AD	AD	X	AD	AD	AD	**AO**	**AO**
5	AD	AD	AD	AD	AD	X	**AO**	AD	**AO**	**AO**
6	AD	AD	AD	AD	AD	TDO	X	AD	**AO**	AD
7	**AO**	TDO	AD	**AO**	AD	AD	AD	X	**AO**	**AO**
8	DO	DO	DO	DO	DO	DO	DO	DO	X	DO
9	AD	DO	AD	DO	DO	DO	AD	DO	**AO**	X

TABLE I
MASKING-HIDDEN DIGIT PAIRS.

- Determining the minimal perceptual color distance between digit segments and background segment which enable to perceive a digit in the digit image (Type 1).
- Evaluating the effect of the number of different colors on the ability to perceive a digit in a digit image (Type 2).
- Determining the boundary between classes of digit images which are geometrically similar to each other (e.g "5" and "9") (Type 3).

The fourth data type included images with the goal of spanning the space of ambiguous samples.

A total of 2500 samples were created and used in the study. Every sample was evaluated by at least 4 subjects. Every sample was shown twice to each subjet. For each sample shown in the user study, the response given by the majority of subjects was considered as the ground-truth label of the sample. Samples marked with "not-a-digit" according to the majority were discarded. The total number of remaining samples with a consistent salient digit was 249, 247, 150, 1477 for the four data types respectively. For further details on the user study, including the experimental setup, creation of the data samples, and resulting statistics of user responses see [13].

VII. Digit-image Classification and Generation using Machine Learning

The goal of this study is to create several Deep Learning based systems to perform the following:

1) Given a digit-image, determine the salient digit ([0,9]).
2) Given a spectral hiding digit-image, determine the masking digit and the hidden digit.
3) Given a masking-hidden digit pair, synthesize a spectral hiding digit-image.

A. Digit-image classification

To determine the salience of each possible digit $[0, 9]$ within a given digit-image, we used our user study data-set to train a fully connected neural network with architecture as detailed in Table II. We will term this network the Salient Digit Classifier (SDC) . Each linear layer, except the last, was followed by a ReLU activation function. The output layer was followed by a Softmax operator. The input corresponded to the 24 RGB values representing the digit image, as defined in Section V. The output layer produced a vector representing the probability of each digit $\in [0, 9]$ appearing in the image. Hyper-parameter values were optimized by random search. Training consisted of 40,000 epochs with learning rate 0.01 and batch size 100. The input data for the network included 4000 samples consisting of the 4 data types of samples labeled in the user study (Section VI) and additional grayscale simple digit images (see Section V). The division of the data-set into training, validation and test sets for each data type is given in Table III. The loss function was the KL-divergence metric [14]. Stochastic Gradient Descent (SGD) [15] was used as optimizer.

Layer id	Layer type	Layer parameters
0	input	size = (24)
1	FullyConnected	size = (24,98)
2	Dropout	(p=0.25)
3	FullyConnected	size = (98,98)
4	Dropout	(p=0.25)
5	FullyConnected	size = (98,49)
6	Dropout	(p=0.25)
7	FullyConnected	size = (49,10)
8	Softmax	size = (10)
Total	4 layers	

TABLE II
Neural network architecture for digit-image classification

Data type	Total size	Training set	Validation set	Test set	Test accuracy
Data type no. 1	249	179	40	30	100%
Data type no. 2	247	178	38	31	96.8%
Data type no. 3	150	109	21	20	85%
Data type no. 4	1477	1037	221	219	90%
Simple samples	1877	1306	280	300	96.3%
Total	4000	2800	600	600	94%

TABLE III
Data set for digit-image classification and test accuracy

Results and discussion: Using 5-fold cross validation, the network achieved on average 96% (std 0.007) accuracy over the validation set and 94% (std 0.008) accuracy over the test set. Accuracy per data type is given in Table III. Confusion matrix per digit is given in Figure 10. Shown values refer to a specific fold with 94% accuracy over test set. These accuracy measures are competitive considering that human subjects did not always agree on the most salient digit.

B. Spectral-hiding digit-image classification

Our second goal was to determine the masking digit and the hidden digit of a spectral-hiding digit-image. To achieve this goal, the trained SDC classification network (Section VII-A) was extended and a transfer learning approach [16] was used. The network's last layer was restructured, to output two labels instead of one. Thus the last layer of the new classifier produced 20 values. The saliency probability distribution of the masking digit was represented in the first 10 values and that of the hidden digit in the remaining 10 values.

The Hyper-parameter values were optimized by random search. Training consisted of 10,000 epochs with a learning rate of 0.005 and batch size 100. The network was trained on spectral-hiding digit image samples that were created following the principles described in Section V. The data-set

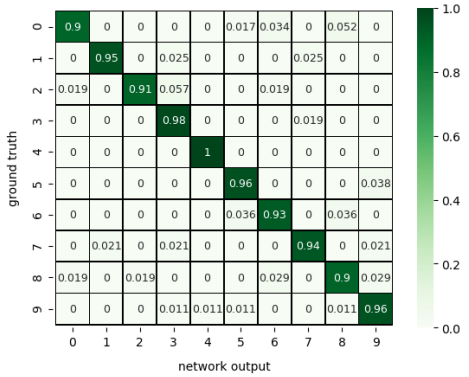

Fig. 10. Confusion matrix of predicted vs ground truth for each digit [0,9]. Total accuracy is 94%.

of 3000 samples was divided into training, validation and test sets with 2000, 500, and 500 samples respectively. SGD [15] was used as the optimizer.

The accuracy measure and the loss function for this network were defined as follows:

Denote by y the true hidden digit and by $\hat{y}$ the network's prediction for the hidden digit. Denote by y_M the true masking digit and by $\hat{y}_M$ the network's prediction for the masking digit. $y, \hat{y}, y_M, \hat{y}_M$ are all vectors of size 10 representing the probability of each digit. $\hat{y}, \hat{y}_M$ are the outputs obtained by applying Softmax after the last layer of the network. According to the assumptions on spectral-hiding image samples (Section V), the hidden digit is well defined and has only one possible salient digit, thus y takes the form of a 1-hot-encoding vector. However the masking digit, y_M, may be perceived as any number of possible salient digits, but must differ from the hidden digit y. Thus the objective function used during training is constructed to concur with these constraints, and is composed of three components:

$$L_H(\hat{y}, y) = \min(KL(\hat{y}, y))$$

where KL is the Kullback–Leibler divergence metric [14]. L_H ensures that the predicted hidden digit $\hat{y}$ is similar to the ground truth hidden digit y.

$$L_{M1} = \min(KL(\hat{y}_M, \bar{y})))$$

where $\bar{y}$ is the complement vector of y. L_{M1} ensures that the predicted masking digit $\hat{y}_M$ differs from the ground truth hidden digit y, thus we want to maximize the distance between $\hat{y}$ and y or minimize the distance to $\bar{y}$.

$$L_{M2} = \min(Entropy(\hat{y}_M))$$

L_{M2} ensures that the predicted masking digit $\hat{y}_M$ has low entropy. This is introduced to prevent the network from converging to a system which predicts $\hat{y}_M$ vectors drawn from a uniform distribution.

The final loss function is the sum of all three components:

$$L(\hat{y}_M, \hat{y}, y) = L_H + L_{M1} + L_{M2}$$

Results and discussion: The trained network achieved 99.8% accuracy over the validation set and 99.8% accuracy over the test set. These measures are high due to the similarity between the original task performed by the SDC Neural Network and the task performed by this network which was trained using transfer learning. Additionally, our relaxed assumption that the masking digit may be classified as any digit except the hidden digit allowed relatively many classification options to be considered as correct.

C. Spectral Hiding Digit Image Generator

The third goal of the research was to build a system that generates spectral-hiding digit-images. To do this, an Auxiliary Classifier GAN (ACGAN) [17] architecture was used. Due to the mode collapse problems in training GAN networks [18] [19], we trained separate ACGAN networks to generate different classes of spectral-hiding digit images.

Generator

Layer id	Layer type	Layer parameters
0	input	size = (120)
1	FullyConnected	size = (120,hidden size)
2	FullyConnected	size = (hidden size,hidden size)
3	FullyConnected	size = (hidden size,24)
Total	4 layers	

Discriminator

Layer id	Layer type	Layer parameters
0	input	size = (24)
1	FullyConnected	size = (24,hidden size)
2	FullyConnected	size = (hidden size,hidden size)
3 real/ fake	FullyConnected	size = (hidden size,1)
3 class	FullyConnected	size = (hidden size,20)
Total	4 layers	

TABLE IV
DETAILED ARCHITECTURE OF THE GENERATOR AND DISCRIMINATOR NETWORKS OF THE ACGAN. HIDDEN SIZE WAS EITHER 200 OR 300.

The separate ACGANs were chosen based on the criterion of pairing digits as described in Section V. Tables V and VI list the different generators (one per row of the table) and the digit pairs that are synthesized by the network.

In all the ACGANs, both the Generator and the Discriminator were fully connected networks. The architecture of the Generator and the Discriminator are detailed in Table IV. Each layer, except the last of both the Discriminator and Generator, was followed by a LeakyReLU activation function with slope of 0.2. The last layer of the Generator network was followed by a Tanh activation function and the last layer of the Discriminator was followed by a Sigmoid activation function for determining the real/fake output and a Softmax activation function for the class output. Input to the generator consisted of a latent noise vector of size 100 sampled from a binary distribution, and two vectors of size 10 representing the hidden digit and the masking digit (represented as one-hot vectors). The hidden layer size was either 200 or 300 dependent on the specific paired digit network. The output of the generator, as well as the input to the discriminator was a vector of size 24 representing a digit-image. The discriminator output included a binary discrimination value (real/fake) as well as two vectors of size 10 representing the predicted hidden digit and masking digit.

The data-set consisted of 7000 spectral-hiding digit images for each pairing option as mentioned in Table I. The number of samples in the training, validation and test set was 5500, 1000 and 500 respectively. These samples were created as described in Section V. Hyper-parameter values were optimized by random search. Training consisted of 1,000 epochs with batch size 100 and learning rate 0.001, β_1 was 0.5 and β_2 was 0.999 for both the Generator and the Discriminator.

Denote by $y, \hat{y}, y_M, \hat{y}_M$ the ground-truth and predicted class of hidden digit and masking digit respectively. All are vectors of size 10. Denote by y_{RF} the true source of the digit image (real data sample / fake generated sample) and by $\hat{y}_{RF}$ the network's prediction for the source of the digit image. y_{RF} is a binary value ($y_{RF} \in \{0, 1\}$) and $\hat{y}_{RF}$ is the output of the sigmoid function that follows the last layer of the network.

Generated digit pairs (masking - hidden)	Loss G	Prediction G	Accuracy G	Loss D	Prediction D	Accuracy D
0-1, 3-1, 8-1, 9-1	1.351	0.387	100%	1.084	0.564	100%
8-3,9-3	1.033	0.428	100%	1.170	0.565	100%
8-4, 9-4	0.921	0.462	100%	1.312	0.522	100%
8-5, 9-5	0.959	0.456	100%	1.231	0.559	100%
0-7, 3-7, 8-7, 9-7	1.262	0.399	100%	1.123	0.622	100%
8-0, 8-2, 8-6, 8-9	1.033	0.426	100%	1.125	0.561	100%
4-1, 7-1, 6-5	1.100	0.409	100%	1.136	0.586	100%

TABLE V
DISAPPEAR ONLY NETWORK RESULTS.

Generated digit pairs (masking - hidden)	Loss G	Prediction G	Accuracy G	Loss D	Prediction D	Accuracy D
2-0, 3-0, 4-0, 5-0, 6-0, 9-0	0.843	0.489	100%	1.299	0.568	100%
2-1, 5-1, 6-1	0.758	0.483	100%	1.296	0.529	100%
0-2, 1-2, 3-2, 4-2, 5-2, 6-2, 7-2, 9-2	0.963	0.442	100%	1.128	0.550	100%
0-3, 2-3, 4-3, 5-3, 6-3	0.910	0.435	100%	1.200	0.558	100%
0-4, 2-4, 3-4, 5-4, 6-4, 7-4	0.961	0.440	100%	1.164	0.532	100%
0-5, 1-5, 2-5, 3-5, 4-5, 7-5	0.863	0.501	100%	1.234	0.603	100%
0-6, 1-6, 2-6, 3-6, 4-6, 7-6, 9-6	0.906	0.405	100%	1.199	0.532	100%
2-7, 4-7, 5-7, 6-7	0.863	0.464	100%	1.230	0.562	100%
0-9, 2-9, 6-9	0.992	0.478	100%	1.085	0.550	100%

TABLE VI
APPEAR & DISAPPEAR NETWORK RESULTS.

The loss function used for training is given by:

$$\begin{aligned} L_{acgan} = &\min\left(\text{Cross Entropy}(\hat{y}, y)\right) \\ &+ \min\left(\text{Cross Entropy}(\hat{y}_M, y_M)\right) \\ &+ \min\left(\text{Binary Cross Entropy}(\hat{y}_{RF}, y_{RF})\right) \end{aligned}$$

Adam [20] was used as an optimizer.

Results and discussion: Examples of the generators' outputs are shown in Figure 11. DO type digit-pairs are shown on the left and the A&D type are shown on the right. Our trained generative networks achieved very good results both in terms of loss, prediction and class accuracy (see Table V) for both generator and discriminator. Further details and network analysis can be found in [13].

VIII. NETWORK PROPERTIES AND PRINCIPLES OF GESTALT

In this section, we show that the trained fully connected SDC neural network detailed in Section VII-A applies some form of Gestalt Principles (see Section II-B) when classifying a digit-image. We do so by running the network on specially designed input samples, chosen to show a specific principle, and analyzing the network output on these samples.

We adopt a perturbation-based approach [21], [22] where perturbation (changes) are applied to an input and the effect on the predicted output is evaluated. We apply perturbations that exemplify specific Gestalt Principles. Consider the example in Figure 12 used in testing for the principle of closure. A digit image (Figure 12a) is perturbed by systematically removing a single digit-segment. The perturbations b-c show

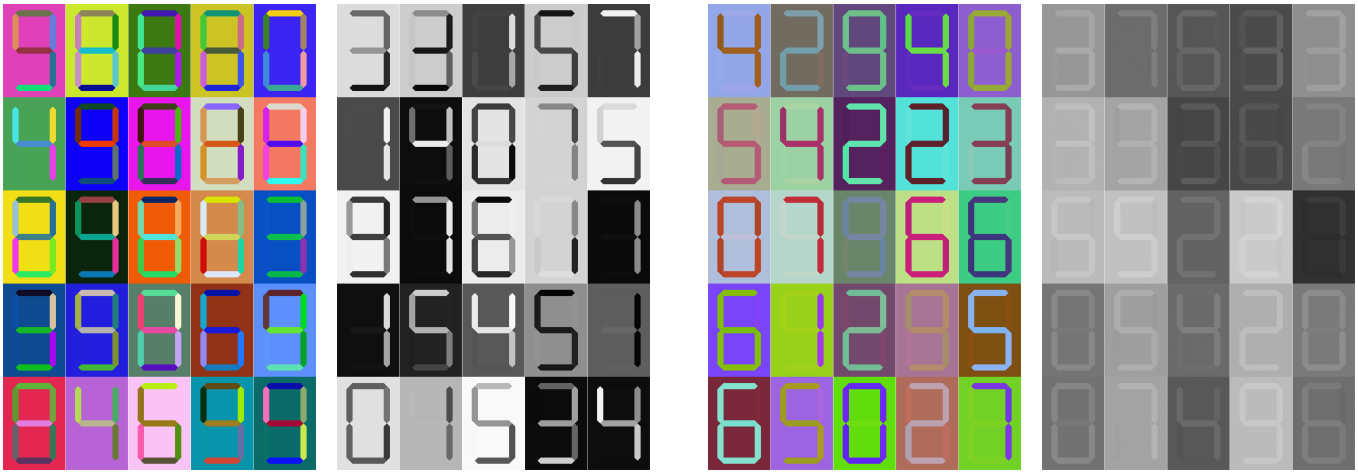

Fig. 11. Samples created by the spectral-hiding digit-image generator. Samples are of type DO (left pair) and of type A&D (right pair). For each pair, images viewed without the spectral filter (left) and the same images as viewed with the spectral filter (right).

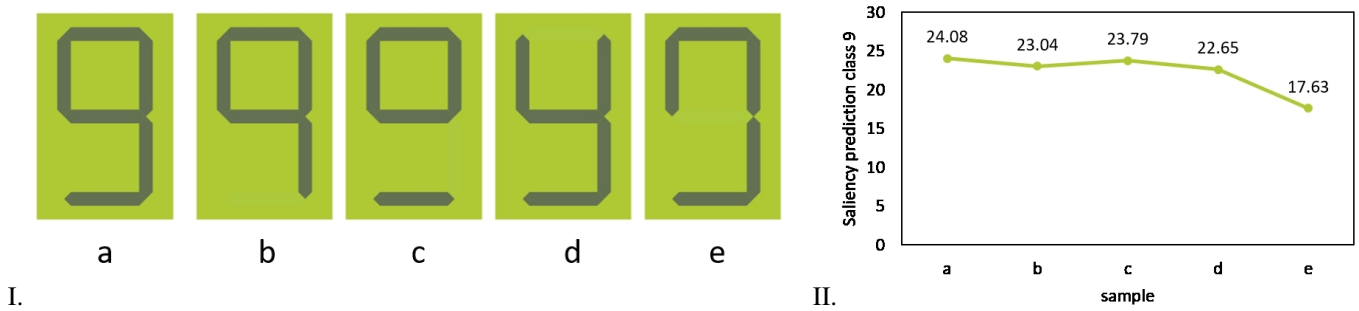

Fig. 12. Perturbation example for closure. I) a) Original digit-image. b-e) Perturbations of image (a) by removing a single digit segment. The closure of the segments in b-c is stronger than in d-e. Due to the lack of closure in d-e, the saliency of digit ”9” is weaker. II) SDC prediction of class ”9” for images in I.

greater closure between segments than the perturbations d-e. We consider the effect of the perturbation on the output produced by the SDC network. Specifically we consider the change in strength of the prediction for digit ”9” as given by the output layer of the SDC (Section VII-A). Figure 12-II plots the prediction strength of the SDC network for the 5 examples of Figure 12-I. Indeed, the samples, b-c, displaying closure show higher prediction values for digit ”9” than samples d-e.

Figure 13 shows pairs of perturbation samples associated with different Gestalt properties. In each pair, Gestalt rules dictate a stronger prediction of the relevant digit for the right sample than for the left sample. This is shown to be true in Figure 14 that plots the strength of the prediction, averaged over 40 digit-image samples of varying color for different digits per each property. Bars show SDC prediction values averaged over samples that show high value of the property (red) and samples with low values of the property (blue). The average relative difference between SDC predictions for high value of the property and low values of the property are 19%, 12% and 39% for proximity, similarity and continuity, respectively, with std of 0.01, 0.088, and 0.036. It is clearly seen that for the examples tested, the network is affected by the Gestalt properties such that the stronger the Gestalt characteristic in the digit-image, the higher the network’s prediction value for the relevant digit.

IX. Discussion

This study deals with hiding data in images using spectral filtering. We developed a system to automatically generate *Spectral Hiding Images* in which a numeric digit is perceived only when viewing the image using a spectral filter (which attenuates high and mid level spectral wavelengths). The hidden digit is masked and can not be seen without the filter.

To achieve this goal three deep neural networks were trained. A classifier was trained to determine the salience of a digit in an image and then extended using transfer learning to evaluate the masking digit and the hidden digit in a spectral hiding image. Finally, a generative model was trained to create diverse spectral hiding digit images. The networks were shown

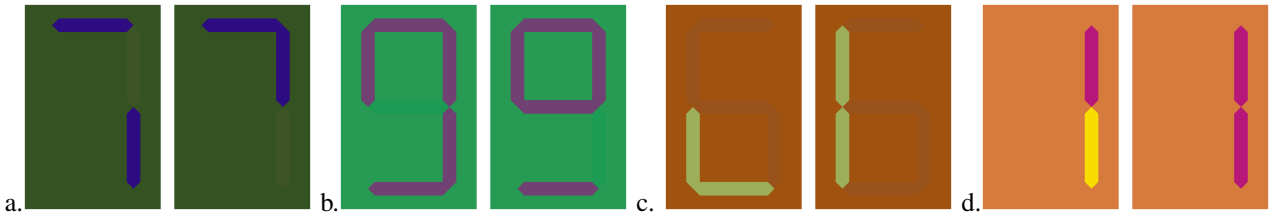

Fig. 13. Inputs to test for network properties. a) Proximity b) Closure c) Continuity d) Color similarity. Gestalt principles dictate that for pairs, the right sample should produce a stronger prediction for the relevant digit than the left sample.

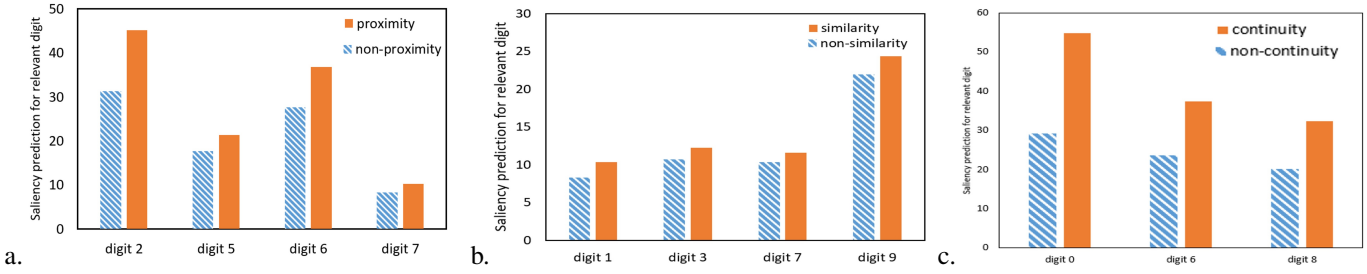

Fig. 14. Concluding graph for network properties. For each digit, network output for samples demonstrating a gestalt principle (Proximity(a), closure(b), continuity(c) and similarity(d)) in comparison to network output for samples demonstrating non-existent relevant gestalt principle are averaged for all test sets.

to perform at a high rate of success. Additionally we showed that several Gestalt principles are expressed in the trained networks' behavior.

Spectral data hiding and the system presented in this study, can be exploited as an added layer of privacy protection, in the case of shoulder surfing, for example when using passwords or in departmentalized environments. A major benefit of the suggested approach is in the ability to mass produce spectral hiding images.

Future work will involve extension of the approach to other types of structured images, which are not necessarily alpha-numeric symbols. In addition, it would be interesting to develop adaptive spectral hiding images that are tuned to different spectral filters and that can be personalized to the individual's visual system.

REFERENCES

[1] S. K. Shevell, "Color mixture," in *The Science of Color*, vol. 1, 2003, pp. 26–29.

[2] T. Smith and J. Guild, "The c.i.e. colorimetric standards and their use," *Transactions of the Optical Society*, vol. 33, no. 3, p. 73, 1931.

[3] M. Wertheimer, "A source book of gestalt psychology." *Psyccritiques*, vol. 13, no. 8, 1968.

[4] L. Nan, A. Sharf, K. Xie, T. Wong, O. Deussen, D. Cohen-Or, and B. Chen, "Conjoining gestalt rules for abstraction of architectural drawings," *ACM Transactions on Graphics (TOG)*, vol. 30, no. 6, pp. 1–10, 2011.

[5] Y. Qi, J. Guo, Y. Li, H. Zhang, T. Xiang, Y. Song, and Z. Tan, "Perceptual grouping via untangling gestalt principles." in *VCIP: Visual Communication and Image Processing*, 2013, pp. 1–6.

[6] "Rgb psychedelic free wallpaper & backgrounds - larutadelsorigens," https://www.larutadelsorigens.cat/wallpaper/iTiixiw_rgb-psychedelic/, accessed: 2020-01-16.

[7] http://www.hidesee.com/demo.html, accessed: 2020-01-16.

[8] J. Qin, Y. Luo, X. Xiang, Y. Tan, and H. Huang, "Coverless image steganography: A survey," *IEEE Access*, vol. 7, pp. 171 372–171 394, 2019.

[9] 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.*, vol. 29, no. 4, pp. 1–51, 2010.

[10] J. Yoon, I. Lee, and H. Kang, "A hidden-picture puzzles generator," in *Computer Graphics Forum*, vol. 27, no. 7, 2008, pp. 1869–1877.

[11] N. J. Mitra, H.-K. Chu, T.-Y. Lee, L. Wolf, H. Yeshurun, and D. Cohen-Or, "Emerging images," *ACM Transactions on Graphics.*, vol. 28, no. 5, pp. 1–8, 2009.

[12] "CIE 1976 L*a*b* colour space–standard," Commission Internationale de L'Éclairage, CIE, Standard, 1976.

[13] H. Shalev, "Shoulder surfing prevention using deep neural networks and spectral filtering," Master's thesis, Dept. of Computer Science, University of Haifa, Haifa, Israel, 2020.

[14] S. Kullback, *Information theory and statistics*. Courier Corporation, 1997.

[15] S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014, ch. 14.

[16] A. Quattoni, M. Collins, and T. Darrell, "Transfer learning for image classification with sparse prototype representations," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[17] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier GANs," in *International Conference on Machine Learning*, vol. 70, 2017, pp. 2642–2651.

[18] T. Che, Y. Li, A. P. Jacob, Y. Bengio, and W. Li, "Mode regularized generative adversarial networks," in *International Conference on Learning Representations (ICLR)*, 2017.

[19] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," in *Advances in neural information processing systems*, 2016, pp. 2234–2242.

[20] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, 2015.

[21] R. C. Fong and A. Vedaldi, "Interpretable explanations of black boxes by meaningful perturbation," in *IEEE International Conference on Computer Vision*, 2017, pp. 3429–3437.

[22] M. T. Ribeiro, S. Singh, and C. Guestrin, ""Why should i trust you?" explaining the predictions of any classifier," in *ACM SIGKDD International onference on knowledge discovery and data mining*, 2016, pp. 1135–1144.