Embedding Dynamic Information in Static Visualization Images with Residual Modeling

Peiying Zhang Chenhui Li *

Abstract

Embedding dynamic information in static visualization images is a practical but challenging problem. It can be used in various applications, such as provide extra data and interactivity through static visualizations. However, embedding large-scale information in a static image may lead to a high decoding error rate. We present a novel framework to implicitly embed dynamic information into a static visualization image while preserving the visual quality of the encoded image. The designed model considers the temporal redundancy within consecutive dynamic information and uses residual modeling in the encoding stage. The decoder network allows users to obtain the dynamic data with a static encoded visualization image. We demonstrate the expressiveness of our model through actual scenarios.

Index Terms: Information visualization; Information steganography; Residual modeling; Autocoding

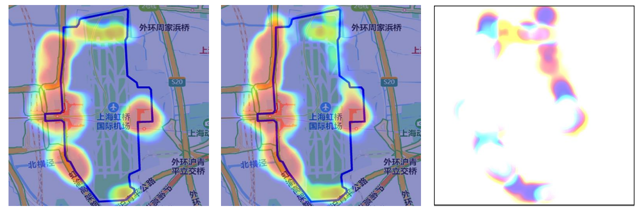
1 Introduction

Data visualizations are widespread in the form of newspapers, posters, academic papers, etc. It is a common way to share and disseminate visual information with a bitmap image on the Internet, instead of the direct transmission of the large-scale raw data. However, a static visualization image can only display limited information. Embedding dynamic information in static visualization images is very meaningful and practical in the field of visualization applications. Although Chen et al. [1] have applied a workflow to explore additional data displayed in Augmented Reality (AR) through a static visualization, it requires the support of additional databases and web servers. As an alternative method, information steganography is more appropriate to hide a large amount of data into static visualization images while preserving their visual perceptual quality.

High-capacity information steganography is a challenging problem because it is easy to generate obvious noise or cause color variations in the encoding stage, or lead to a high reconstructed error rate in the decoding stage. Previous studies focus on image steganography [2, 5]. It has unsatisfied performance to directly use image steganography model trained on the natural images dataset to solve the dynamic information steganography problem. On the one hand, it ignores the temporal redundancy within consecutive dynamic frames. There is no need to encode the overlapping parts several times. On the other hand, the features of residuals are quite distinct from natural images. For example, the residuals are highly sparse, as shown in Fig. 1. Therefore, we utilize residual modeling to hide the secret dynamic information in the static visualization image.

Fig. 2 shows our pipeline for embedding dynamic information in static visualization images. First, the original dynamic information is converted to the residual type, regarding the static container image as reference. Next, an encoder network embeds secret dynamic residuals in the static container image and outputs a static coded image. Users can share coded images with others through digital transmission. When a user wishes to obtain the dynamic information hidden in the static image, the encoded image can be uploaded to the decoder network. After data recovery, the user receives the decoded information.

*Peiying Zhang and Chenhui Li are with the School of Computer Science and Technology, East China Normal University. The authors wish to acknowledge the support from NSFC under Grants (No. 61802128). e-mail: (zhangpeiying17@gmail.com, chli@cs.ecnu.edu.cn).



(a) Frame_i (b) Frame_{i+1} (c) Residuals

Figure 1: Sample residual of dynamic information frames. The residuals are highly sparse.

2 Method Description

As shown in Fig. 2, given a set of dynamic visualization frames $S_F = \{S_{F_1}, S_{F_2}, ..., S_{F_k}\}$, we first convert the plain secret images to the residuals $S_R = \{S_{R_1}, S_{R_2}, ..., S_{R_k}\}$, which takes the carrier (also called container or cover) image C_I as reference. Then the encoder network embeds the dynamic message in residual type S_R to the carrier image C_I and outputs the resulting coded image $C_I{'}$. Users can share coded images with others through digital transmission. The decoder network $Dec()$ retrieves the dynamic information $S_R{'}$ from the static coded image $C_I{'}$. After adding the reconstructed residuals $S_R{'}$ to the reference image, we can obtain the revealed dynamic information $S_F{'}$. Our goal is to learn functions $Enc()$ and $Dec()$ such that the perceptual differences between $C_I{'}$ and C_I are small according to human vision, while the recovered information $S_F{'}$ is accurate as the original S_F.

2.1 Residual Modeling

As mentioned above, the residuals are highly sparse and it requires less effort to embed the high-sparse information into a static visualization image compared with the original secret dynamic frame. Therefore, we utilize residual modeling to hide the secret dynamic information in the static visualization image. Specifically, the static carrier image C_I is regarded as the reference image in the encoding stage, and the encoded image $C_I{'}$ is regarded as the reference image in the decoding stage. Addition and subtraction operations are applied to the secret dynamic frames and the carrier image to obtain the residuals.

2.2 Encoder and Decoder

As demonstrated in UDH [4], the encoder network transform the images with low-frequency content to the high-frequency repetitive patterns. Inspired by UDH, we feed only secret residuals to the encoder network instead of concatenating the feature vectors of secret messages and carrier visualization image, which is a common

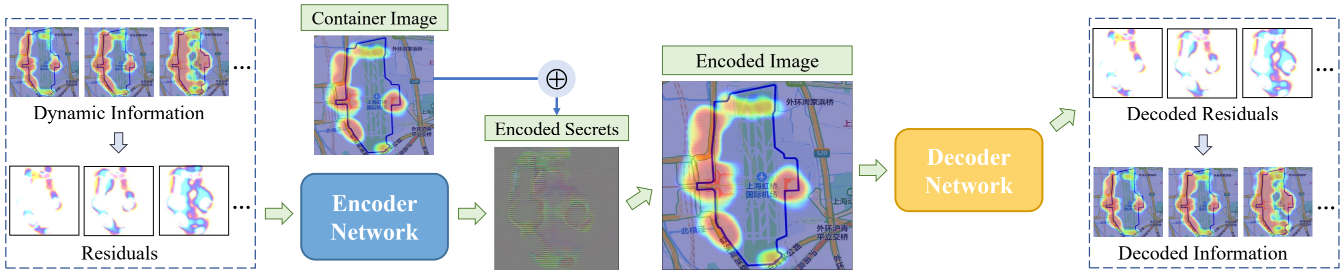


Figure 2: The structure of our method. First, the original dynamic information is converted to the residual type, regarding the static container image as reference. Next, an encoder network embeds secret dynamic residuals in the static container image and outputs a static coded image. Users can share coded images with others through digital transmission. When a user wishes to obtain the dynamic information hidden in the static image, the encoded image can be uploaded to the decoder network. After data recovery, the user receives the decoded information.

way in image steganography. After obtaining the encoded image of secret residuals, we add it to the cover visualization image directly. Since the encoded secret image is with high-frequency property and the human visual system is insensitive to the high-frequency content, perturbations in the encoded image $C_I{}'$ have limited influence. The decoder network is trained to be only sensitive to high-frequency content. After residual correction, we can obtain the reconstructed dynamic information. We use the joint loss to train the encoder and decoder network:

$$L_{\text{joint}} = L_{Enc} + \lambda L_{Dec} = \left\| C_I - C_I{}' \right\|_F^2 + \lambda \left\| S_F - S_F{}' \right\|_F^2 \qquad (1)$$

where S_F is the original dynamic visualization frames and $S_F{}'$ is the revealed information of the decoder network. C_I is the original carrier image and $C_I{}'$ is the coded image of the encoder network. $\| \cdot \|_F^2$ represents the Frobenius norm. λ is the weight coefficient to balance the visual quality of the encoded images and the decoding accuracy. Higher λ results in higher decoding accuracy and decreased perceptual quality. During the training process, we co-train the encoder network and decoder network with the ADAM optimizer. In the testing process, the encoder and decoder can be used separately.

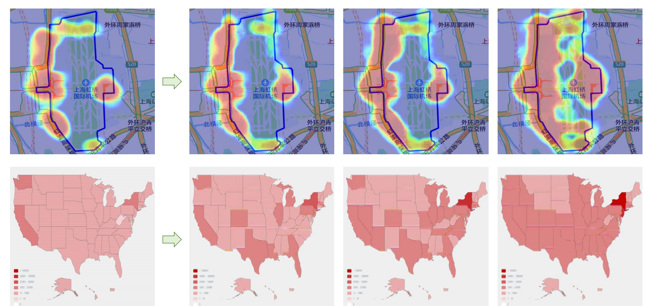
3 Results

Our method for embedding dynamic information in static visualizations can be used for a number of applications. We show the effect of our method on two real-world datasets: passenger flow data of Hongqiao Airport from Tencent Bigdata and public epidemic data of COVID-19. As shown in Fig. 3, the encoded image (Fig. 3a) looks identical to the original chart image since the perturbations in the coded image have limited influence on human perception. The reconstructed secret residuals (Fig. 3b-d) are with a low decoding error rate, which allow users to explore the additional dynamic data.

To evaluate the performance of our method, we follow the steganography indices settings of VisCode [5]. We use three metrics: the peak signal-to-noise ratio (PSNR), the SSIM [3], and the learned perceptual image patch similarity (LPIPS) [6]. The PSNR metric is widely used for evaluating image distortion. The SSIM metric is useful to measure the structural similarity between two images. The LPIPS metric can measure the perceptual similarity by modeling human judgments. Higher PSNR, higher SSIM, and lower LPIPS are better. As shown in Table 1, the result demonstrates that VisCode can encode and decode dynamic information while preserving the perceptual quality of the various types of static visualizations.

4 Conclusion

In this poster, we propose a method for embedding dynamic information in static visualization images with residual modeling. In the future, we plan to collaborate with experts to improve the workflow and user experience of exploring dynamic information with a static image.



(a) Encoded Image (b) Frame$_1$ (c) Frame$_2$ (d) Frame$_3$

Figure 3: Perceptual image quality of the static encoded image and decoded dynamic information.

Table 1: Evaluation results of steganography indices

Network	PSNR	SSIM	LPIPS
Encoder	37.33	0.961	0.006
Decoder	30.83	0.907	0.097

References

[1] Z. Chen, W. Tong, Q. Wang, B. Bach, and H. Qu. Augmenting static visualizations with paparvis designer. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–12, 2020.

[2] J. Fu, B. Zhu, W. Cui, S. Ge, Y. Wang, H. Zhang, H. Huang, Y. Tang, D. Zhang, and X. Ma. Chartem: Reviving chart images with data embedding. *IEEE Transactions on Visualization and Computer Graphics*, 2020.

[3] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[4] C. Zhang, P. Benz, A. Karjauv, G. Sun, and I.-S. Kweon. Udh: Universal deep hiding for steganography, watermarking, and light field messaging. In *34th Conference on Neural Information Processing Systems, NeurIPS 2020*. Conference on Neural Information Processing Systems, 2020.

[5] P. Zhang, C. Li, and C. Wang. Viscode: Embedding information in visualization images using encoder-decoder network. *IEEE Transactions on Visualization and Computer Graphics*, 2020.

[6] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018.