Stealthy Location Tracking with Ninja Codes

Yusuke Imoto
Osaka University
Japan
yusuke.imoto@sens.sys.es.osaka-u.ac.jp

Shunya Kato
Kyoto University
Japan
s-kato@nlp.ist.i.kyoto-u.ac.jp

Yuichiro Takeuchi
Sony CSL Kyoto
Japan
Wikitopia Institute
Japan
yutak@acm.org

ABSTRACT

In this paper we describe Ninja Codes, neurally-generated fiducial markers that blend naturally into real-world environments. Taking inspiration from recent advances in deep steganography, we train a neural network that transforms arbitrary images into functional fiducial markers (Ninja Codes) with minimal visual changes. Ninja Codes can be pasted onto various surfaces, to provide location tracking capability for applications such as augmented reality, robotics, etc. Ninja Codes can be printed using common color printers on regular paper, and can be detected by any hardware equipped with a standard RGB camera and capable of running inference.

CCS CONCEPTS

• **Human-centered computing** → **Ubiquitous and mobile computing systems and tools**; • **Computing methodologies** → **Computer vision**.

KEYWORDS

Ninja Codes, fiducial markers, location tracking, deep steganography, neural networks

ACM Reference Format:
Yusuke Imoto, Shunya Kato, and Yuichiro Takeuchi. 2024. Stealthy Location Tracking with Ninja Codes. In *SIGGRAPH Asia 2024 Emerging Technologies (SA Emerging Technologies '24), December 03-06, 2024.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3681755.3688953

1 INTRODUCTION

Fiducial markers (Figure1, left), i.e., printable images whose precise locations can be detected via computer vision, is an inexpensive and reliable location tracking solution that is used for many applications including augmented reality, robotics, and motion-based user interfaces. However, due to their typically conspicuous appearances, and the fact that a large number of markers need to be placed throughout the environment to provide robust location tracking, their utility is limited in environments where aesthetic concerns carry weight such as residential homes or city streets.

This paper describes Ninja Codes (Figure 1, right), a new class of fiducial markers that blend seamlessly into various environments. Building on prior work in the field of deep steganography [Baluja 2017], we use a neural network to introduce visual alterations to arbitrary images, that seem subtle and discreet to human eyes but make the images appear as fiducial markers to trained detector networks. By generating Ninja Codes from photographs of real-world surfaces such as wooden tabletops or concrete walls, we can add location tracking capabilities to environments in visually unobtrusive ways. The codes can be printed on regular paper using standard, off-the-shelf printers, and can theoretically be embedded into surface patterns of textiles, 3D printed objects, etc. Detection of Ninja Codes can be performed by any device with an RGB camera and sufficient processing power to run inference.

With further refinements, we believe that Ninja Codes can serve as a key enabling technology leading to a wave of new applications that utilize location tracking in novel environments.

Figure 1: A conventional fiducial marker (left). Ninja Code: a discreet neural fiducial marker (right).

2 RELATED WORK

Fiducial markers such as ARTags [Fiala 2005] have been used for location tracking since the early 1990s. The markers are typically black-and-white, and are similar in appearance to QR Codes but visually simpler due to the smaller amounts of encoded data. The vast majority of fiducial markers have handcrafted design schemes and are detected using CPU-based algorithms, but some recent work have explored the use of deep learning to create fiducial markers with specialized characteristics [Yaldiz et al. 2021].

Deep steganography refers to the use of deep learning techniques to conceal arbitrary messages within images. Examples of work in this area include HiDDeN [Zhu et al. 2018] and StegaStamp [Tancik et al. 2020]; the resulting images show minimal signs of tampering that are virtually imperceptible to the human eye. Our work extends such prior art, modifying existing techniques to render images that provide location tracking functions in addition to hiding data.

3 NINJA CODES

Figure 2 shows the end-to-end architecture that we use to train the networks involved in the creation and detection of Ninja Codes. The

SA Emerging Technologies '24, December 03-06, 2024, Tokyo, Japan

ACM ISBN 979-8-4007-1137-4/24/12.
https://doi.org/10.1145/3681755.3688953

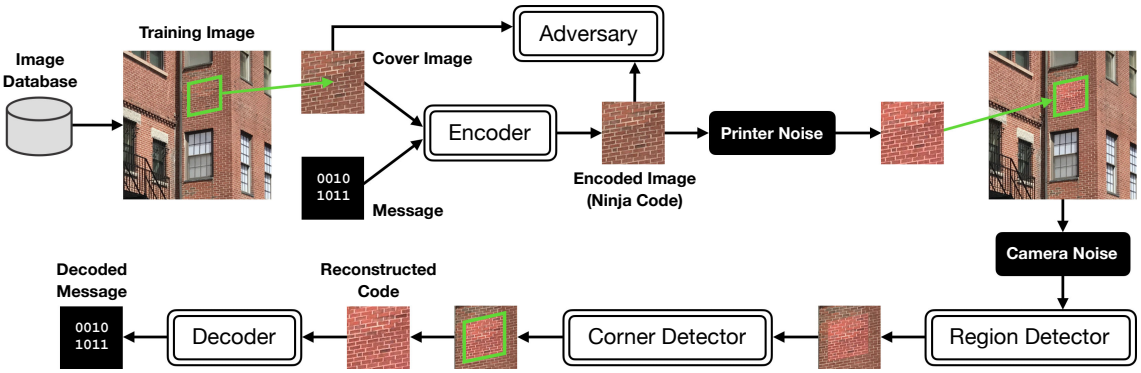

Figure 2: Ninja Codes training architecture. Five models are trained simultaneously: encoder, region detector, corner detector, decoder, and adversary. Noise functions are applied to simulate perturbations arising from printing and camera capture. The loss function minimizes the perceptual differences between the cover and encoded images, and detection/decoding errors.

encoder is based on U-Net [Ronneberger et al. 2015], which takes an arbitrary RGB image (cover image) and a 36-bit binary message as input and produces an encoded image. The region detector (based on SSD [Liu et al. 2016]) takes a camera image and predicts square regions that are likely to contain Ninja Codes, which are in turn given to the corner detector that finds the four corners (and thus exact contours) of the Ninja Codes. Finally, the decoder recovers binary messages from the detected codes. Additionally, an adversary network is introduced to the training process, which attempts to distinguish between encoded and unencoded images.

Provided that code sizes are standardized, the 6 DoF (degrees of freedom) position and pose of the device's camera can be calculated from the Ninja Code contours, allowing the codes to function as a location tracking solution.

Ninja Codes are meant to be printed onto physical media (such as paper), and captured in the wild using cameras; both of these processes inevitably introduce color shifts and other perturbations. Noise modules are added to the training process, to simulate such perturbations and create codes with real-world robustness.

Training was done on a workstation with 2 NVIDIA RTX 3090 GPUs. As the loss function we used a weighted sum of image loss (difference between cover/encoded images), regression/classification losses (region detection error), corner loss (corner detection error), message loss (difference between original/recovered messages), and adversary loss (the degree to which the adversary could distinguish encoded images). As training data we used a total of 48,000 images, of which 43,300 were taken from the COCO dataset [Lin et al. 2014] and 4,700 were taken from the DTD dataset [Cimpoi et al. 2014].

Figure 3 shows several examples of generated Ninja Codes. Despite noticeable artifacts, the visual characteristics of the cover images are well preserved. In our experiments where we printed out Ninja Codes and measured their detection performance in a well-lit residential room, we found that the codes can be reliably detected, albeit at lower accuracies compared to conventional ARTags.

Figure 3: Examples of cover images and Ninja Codes.

4 CONCLUSION AND FUTURE WORK

In this paper we described Ninja Codes, neurally generated fiducial markers that blend into real-world environments. While additional work is needed to further suppress artifacts, the results show promise in realizing discreet fiducial markers that can offer reliable location tracking in varied real-world environments.

REFERENCES

Shumeet Baluja. 2017. Hiding Images in Plain Sight: Deep Steganography. In *Proceedings of NeurIPS 2017*. 2066–2076.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. 2014. Describing Textures in the Wild. In *Proceedings of CVPR 2014*. 3606–3613.

Mark Fiala. 2005. A Fiducial Marker System Using Digital Techniques. In *Proceedings of CVPR 2005*. 590–596.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. 2014. Microsoft COCO: Common Objects in Context. In *Proceedings of ECCV 2014*. 740–755.

Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Cheng-Yang Fu Scott Reed, and Alexander C. Berg. 2016. SSD: Single Shot MultiBox Detector. In *Proceedings of ECCV 2016*. 21–37.

Olaf Ronneberger, Philipp Fischer, and Thomaas Brox. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Proceedings of MICCAI 2015*. 234–241.

Matthew Tancik, Ben Mildenhall, and Ren Ng. 2020. StegaStamp: Invisible Hyperlinks in Physical Photographs. In *Proceedings of CVPR 2020*. 2117–2126.

Mustafa B. Yaldiz, Andreas Meuleman, Hyeonjoong Jang, Hyunho Ha, and Min H. Kim. 2021. DeepFormableTag: End-to-end Generation and Recognition of Deformable Fiducial Markers. *ACM Transactions on Graphics* 40, 4, Article 67 (2021).

Jiren Zhu, Russel Kaplan, Justin Johnson, and Fei-Fei Li. 2018. HiDDeN: Hiding Data with Deep Networks. In *Proceedings of ECCV 2018*. 657–672.