APPLIED RESEARCH

Deep Learning for Predictive Analytics in Reversible Steganography

CHING-CHUN CHANG [1], **XU WANG** [2,3], **SISHENG CHEN** [2,4], **ISAO ECHIZEN** [1], **VICTOR SANCHEZ** [5], **AND CHANG-TSUN LI** [6]

[1]National Institute of Informatics, Tokyo, Japan
[2]Department of Information Engineering and Computer Science, Feng Chia University, Taichung, Taiwan
[3]School of Information Science and Engineering, University of Jinan, Jinan, China
[4]School of Big Data and Artificial Intelligence, Fujian Polytechnic Normal University, Fuzhou, China
[5]Department of Computer Science, University of Warwick, Coventry, U.K.
[6]School of Information Technology, Deakin University, Geelong, VIC, Australia

Corresponding author: Ching-Chun Chang (ccchang@nii.ac.jp)

ABSTRACT Deep learning is regarded as a promising solution for reversible steganography. There is an accelerating trend of representing a reversible steo-system by monolithic neural networks, which bypass intermediate operations in traditional pipelines of reversible steganography. This end-to-end paradigm, however, suffers from imperfect reversibility. By contrast, the modular paradigm that incorporates neural networks into modules of traditional pipelines can stably guarantee reversibility with mathematical explainability. Prediction-error modulation is a well-established reversible steganography pipeline for digital images. It consists of a predictive analytics module and a reversible coding module. Given that reversibility is governed independently by the coding module, we narrow our focus to the incorporation of neural networks into the analytics module, which serves the purpose of predicting pixel intensities and a pivotal role in determining capacity and imperceptibility. The objective of this study is to evaluate the impacts of different training configurations upon predictive accuracy of neural networks and provide practical insights. In particular, we investigate how different initialisation strategies for input images may affect the learning process and how different training strategies for dual-layer prediction respond to the problem of distributional shift. Furthermore, we compare steganographic performance of various model architectures with different loss functions.

INDEX TERMS Deep learning, modularity, predictive analytics, reversible steganography.

I. INTRODUCTION

Steganography is recognised as an important research field in multimedia forensics and security. It is the practice of concealing a message within a *cover* object to produce a *stego* object [1]. Steganographic technology has a wide range of applications, including but not limited to covert communications [2], ownership identification [3] and tamper proofing [4]. Reversible steganography is used primarily for data authentication in fidelity-sensitive applications such as

The associate editor coordinating the review of this manuscript and approving it for publication was Ahmed Farouk [ID].

creative production, forensic science, legal proceedings, medical diagnosis and military reconnaissance [5], [6], [7], [8]. It lightens the burden of metadata management by invisibly inserting auxiliary authentication information (e.g. a tamper-proof fingerprint, a hash digest or a digital signature) into the data. Message embedding is carried out in a reversible manner; that is, steganographic distortion can be removed and the original data reconstructed in a time-reversed fashion. The ability to reverse a steganographic process, namely *reversibility*, is a desirable feature in this era of data-centric artificial intelligence. Many real-world intelligent systems require colossal amounts of data to be collected and used in the

training and validation processes. However, hostile attacks may occur during data communications. It has been reported that learning machinery is susceptible to data-driven adversarial attacks [9], [10], [11]. For instance, adversarial perturbation is an invisible noise crafted to fool a deep-learning system. Another example is adversarial contamination that imperceptibly injects poisonous samples into the training pool, thereby compromising the accuracy of the system. To combat such cybersecurity threats, steganography-based authentication schemes can be applied to verify the integrity of training and validation data, thereby ensuring secure data exchange. If an irreversible steganographic algorithm is incorporated into a data exchange protocol for authentication purposes, steganographic distortion might present uncontrollable risks to the reliability of data-centric autonomous machines [12]. Therefore, the ability to remove steganographic distortion and restore data integrity is of paramount importance.

The core of reversible steganography, in common with lossless compression, is predictive coding [13]. Prediction-error modulation is one of the main pillars of contemporary reversible steganography due to its optimum rate-distortion performance [14], [15], [16]. A prediction-error modulation scheme has an analytics module to serve the purpose of context-aware pixel intensity prediction, and a coding module to perform encoding/decoding in the residual domain. Deep learning is currently at the very heart of multimedia analytics and it offers appealing solutions for pixel intensity prediction. In this paper, we investigate the cause-effect relationships between the variables of interest regarding the use of deep learning models for pixel intensity prediction. In particular, we investigate the impacts of different initialisation and training strategies upon predictive accuracy. Our contributions are summarised as follows:

- We compare zero initialisation and local-mean initialisation to gain an insight into whether pre-processing of input images with a local-mean filter can help the models perform better or has a counterproductive effect.
- We explore universal training, independent training, and causal training to gain an insight into the problem of distributional shift in dual-layer prediction caused by steganographic distortion.
- We carry out an ablation study to analyse the impacts of different loss functions upon steganographic performance with the aim of providing insights regarding the applicability of neural network models originally proposed for low-level computer vision, and demonstrate the state-of-the-art steganographic performance with the residual dense network (RDN) which is an advanced model for image super-resolution [17] and image restoration [18].

The remainder of this paper is organised as follows. Section II provides a literature review on the recent development of steganography with deep learning. Section III delineates modules of a reversible steganographic scheme. Section IV introduces the methodology concerning intensity initialisation strategies, dual-layer training strategies, and neural network models. Section V evaluates the impact of different variables of interest upon predictive accuracy and steganographic performance. Limitations of this study are discussed in Section VI. Conclusions are drawn in Section VII.

II. LITERATURE REVIEW

Artificial intelligence has been evolving over the years, being at the forefront of transformations of the world we live in [19]. Deep learning, as a new class of multi-purpose intelligent algorithms, learns how to solve complicated tasks through the observation of a large amount of data, and has brought about groundbreaking advances in many branches of science [20]. The foundations of deep learning are neural networks, or connectionist systems, which are capable of discovering intricate structures in high-dimensional data via multiple layers of artificial neurons. The recent development of steganographic methods is centred on the use of deep-learning models to enhance performance. Significant breakthroughs have been achieved in some basic properties such as *capacity* (the allowable size of the embedded message) and *imperceptibility* (the perceptual quality of the stego object), as well as some application-oriented properties such as *secrecy* (the degree to which the stego object can elude detection), and *robustness* (the degree to which the embedded message can survive distortions of various forms) [21]. In this section, we briefly review some seminal studies of secure and robust steganography with deep learning and then discuss the use of neural networks in reversible steganography with a contrast drawn between end-to-end and modular frameworks. Note that our study focuses on the basic properties (i.e. capacity and imperceptibility) and reversibility since application-oriented requirements can be contradictory and difficult to achieve concurrently.

A. SECRECY

Secrecy is at the heart of covert communications. The ability to pass secret information under surveillance is essential to espionage and military operations. It has been reported that deep learning can be used to identify locations in a cover image at which message embedding would not arouse suspicion [22], [23], [24]. More technically, a deep learning algorithm assigns a cost to every pixel to quantify the effect of making modifications. In this way, a high degree of perceptual and statistical undetectability can be reached by minimising the cost. Other approaches include using generative models to convert a given message into a cover image which is less vulnerable to steganalysis [25], to transform a cover image along with a secret message into a stego image [26], and to encode a message directly into a realistic stego image in the absence of an explicit cover image [27].

B. ROBUSTNESS

Robustness is a prioritised requirement for copyright protection. In commercial applications, entrepreneurs can prevent

copyright infringement by embedding a registered watermark into digital assets. Deep learning has been used to embed invisible watermarks into images and videos in a durable way in order to identify copyright ownership and deter illegitimate copying [28]. Unauthorised screen recording is a new scenario in which an adversary attempts to capture an electronically displayed still photograph or video footage with a digital camera, causing optical interference in watermark extraction. It has been shown that neural networks can be trained to simulate optical interference and then learn to encode and decode watermarks in a robust manner [29]. Decoding accuracy can also be enhanced by using neural networks to remove light artefacts such as moiré fringes, a type of textile with a rippled or wavy appearance, prior to watermark extraction [30]. To facilitate augmented reality, deep learning has been used to decode hyperlinks embedded in physical photographs rather than digital media, subject to real-world variations in print quality, illumination, occlusion and viewing distance [31].

C. REVERSIBILITY—END-TO-END FRAMEWORK

Reversibility is a desirable characteristic for applications in which accuracy and consistency of data are important. It is a requisite for preventing an accumulation of steganographic distortion over each data transmission. A reversible steganographic method often relies on intricate logical operations to regulate imperceptibility and guarantee reversibility. Such operations are hard to achieve using existing neural network models. One of the earliest approaches uses a U-shaped network (U-Net) to automatically encode a cover image and a secret message into a stego image and a separate neural network model for decoding [32]. Another approach encodes a message bitstream into a cover image via a generative model, trains a cycle-consistent generative adversarial network (CycleGAN) to learn the back-and-forth transformation between the cover and stego images, and extracts the message bitstream from the stego image by using another neural network model [33]. Furthermore, an intriguing study shows that both encoding (forward mapping) and decoding (inverse mapping) can be performed by using a single invertible neural network (INN) [34]. While a monolithic end-to-end system generally offers a large steganographic capacity, a common limitation within this type of framework is imperfect reversibility due to the presence of an information bottleneck (i.e. a form of lossy compression) in neural networks. The lack of transparency, interpretability and explainability in neural networks adds an extra dimension to the problem.

D. REVERSIBILITY—MODULAR FRAMEWORK

Although reversibility is not an all-or-nothing proposition, the unreliability of end-to-end learning may pose uncontrollable risks in certain circumstances. Problem decomposition or modularisation is essential to addressing complex problems. Existing reversible steganographic schemes often consist of *coding* and *analytics* modules. In general, the coding module handles the encoding and decoding mechanisms, which are

designed to achieve perfect reversibility subject to an imperceptibility constraint. The analytics module models the data distribution and exploits data redundancy in order to optimise steganographic rate-distortion performance. For instance, the regular–singular (RS) scheme introduced by Fridrich et al. constructs a discrimination function to categorise image blocks into regular, singular and unusable groups on the basis of the smoothness prior, and uses invertible flipping of the least significant bits to achieve invisible and erasable embedding [35]. We can define the former part as the analytics module and the latter part as the coding module. The problem of limited reversibility in previous deep-learning approaches lies in the difficulty of lossless coding in an end-to-end learning fashion. A recent study achieves perfect reversibility with a deep-learning-based RS scheme following this established *modular* framework [36]. A conditional generative adversarial network (GAN), referred to as pix2pix [37], is applied to improve the performance of the discrimination function. It has been shown that by partitioning a scheme into coding and analytics modules and deploying neural networks in the latter, reversibility can be reliably guaranteed.

E. ANALYSIS OF PRIOR ART

A combination of prediction-error modulation with deep learning has been described in two studies [38] and [39]. Neural networks are incorporated into the schemes based upon *modularity*. The most notable feature in common is that both studies divide pixels into *query* and *context* sets by using a chequered pattern and develop neural network models for predicting the query from the context. The main difference is neural network architectures. The former constructs a multi-scale convolutional neural network (MS-CNN) to address the issue of restricted receptive fields in traditional predictors. The latter applies a persistent memory network (MemNet) originally developed for image denoising and image super-resolution [40]. From a certain perspective, context-aware pixel intensity prediction is closely related to *low-level computer vision* tasks such as image denoising and image super-resolution because they all rely largely on low-level (or pixel-level) features such as edges, contours and textures, rather than high-level semantics [41]. Therefore, it is expected that advanced neural networks from the low-level computer vision domain can be applied directly with minor modifications. Another difference is the setting of the initial intensities of the query pixels. The former simply sets the query pixels to zero and learns to predict from scratch. By contrast, the latter initialises them to the mean of neighbouring pixels and performs prediction in a coarse-to-fine manner. Consequently, a problem of pixel-intensity prediction is, in a certain sense, transformed into a problem of image denoising or image super-resolution. Last but not least, the chequered pattern for context/query splitting features a dual-layer prediction: pixels assigned as the query in the first round, after partial embedding, can be assigned as the context in the second round. Such a practice gives rise to the problem of *distributional shift*—the data distribution in
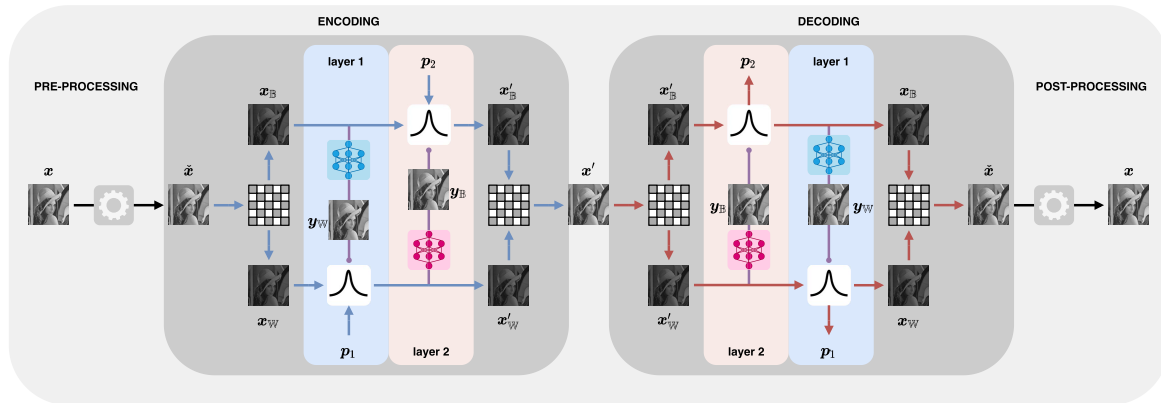
FIGURE 1. Schematic workflow of prediction-error modulation. From left to right: pre-processing, encoding, decoding and post-processing stages.

the target (online) domain for deployment shifts from that in the training (offline) domain. In consequence of the distortion introduced in the preceding steganographic process, assigning those distorted pixels as the context could introduce discrepancy and bias into the contextual information, thereby undermining predictive performance. The expected performance relies on the extrapolation (generalisation) capability of neural networks. To summarise, this work aims to seek answers to the following questions:

- Can different intensity initialisation strategies affect predictive accuracy of neural networks substantially?
- Can the problem of distributional shift be mitigated, thereby enabling efficient dual-layer prediction?
- Can off-the-shelf neural networks from the low-level computer vision domain be transferred to perform pixel intensity prediction?

TABLE 1. List of variables and operations.

var. & op.	definition
x	cover image
x'	stego image
$\tilde{x}$	processed image
$x_{\mathcal{Q}}$	initialised query pixels
$y_{\mathcal{Q}}$	predicted query pixels
ε	prediction errors
ε'	modulated prediction errors
v	overflow-status register
m	message
p	payload
t	index for v and p
ϑ	stego-channel parameter
$\mathbb{B}$	black set of pixel coordinates
$\mathbb{W}$	white set of pixel coordinates
net_1	neural network for first-layer prediction
net_2	neural network for second-layer prediction
$\text{scale}_{\Downarrow}$	downscaling of possibly overflowing pixels
$\text{scale}_{\Uparrow}$	upscaling of possibly overflowing pixels
split	context-query splitting
merge	context-query merger
encode	prediction-error modulation (payload embedding)
decode	prediction-error de-modulation (payload extraction)
concat	concatenation of bit-streams
deconcat	de-concatenation of bit-streams

III. REVERSIBLE STEGANOGRAPHY

In this section, we delineate a reversible steganographic scheme based on prediction-error modulation. We begin with an overview of the scheme and then dissect each scheme component.

A. SCHEME OVERVIEW

A schematic workflow of the encoding and decoding procedures is provided in Figure 1 with the definitions of variables and operations listed in Table 1. The scheme based on prediction-error modulation can be broken down into an analytics module and a coding module. The former models the distribution of natural images and predicts the intensity of query pixels on the basis of the available context pixels. The latter encodes/decodes messages into/from prediction errors. Consider the following scenario: A sender (encoder) embeds a message m into a cover image x to produce a stego image x' and then transmits it to a receiver (decoder) who extracts the message and restores the cover image on receipt of the stego image.

At the encoder side, we pre-process the cover image to prevent pixel intensity overflow during message embedding at the cost of generating extra auxiliary information. We treat the auxiliary information as a part of the payload and concatenate it with the intended message. Then, a chequered pattern is applied to split the image into a black set and a white set of pixels, denoted respectively by $x_{\mathbb{B}}$ and $x_{\mathbb{W}}$. In the first-layer prediction, we designate the black set as the context set and the white set as the query set. The roles are reversed in the second-layer prediction. We make use of a neural network or a predictive algorithm to estimate the intensities of the query pixels based on the observed context pixels. The prediction errors or residuals, denoted by ε, are computed by subtracting the predicted values from the actual values. A portion of the payload is embedded in the residual domain through arithmetic operations. The modulated prediction errors, denoted by ε', are then added to the predicted values, resulting in a slight distortion to the query pixels. The process can be

Algorithm 1 Pre-Processing

Input: x, ϑ
Output: $\check{x}, v$
  ▷ down-scaling
  $v = \varnothing$
  for $i \leftarrow 1$ to height(x) **do**
    for $j \leftarrow 1$ to width(x) **do**
      $\check{x}_{i,j} \leftarrow x_{i,j}$
      if $x_{i,j} \geq 255 - \vartheta + 1$ **then**   ▷ overflow
        $\check{x}_{i,j} \leftarrow x_{i,j} - \vartheta$
        $v = \text{concat}(v, 1)$
      else if $255 - \vartheta \geq x_{i,j} \geq 255 - 2\vartheta + 1$ **then**
        $v = \text{concat}(v, 0)$
      else if $x_{i,j} \leq 0 + \vartheta - 1$ **then**   ▷ overflow
        $\check{x}_{i,j} \leftarrow x_{i,j} + \vartheta$
        $v = \text{concat}(v, 1)$
      else if $0 + \vartheta \leq x_{i,j} \leq 0 + 2\vartheta - 1$ **then**
        $v = \text{concat}(v, 0)$

Algorithm 2 Post-Processing

Input: $\check{x}, v, \vartheta$
Output: x
  ▷ up-scaling
  $t = 1$
  for $i \leftarrow 1$ to height($\check{x}$) **do**
    for $j \leftarrow 1$ to width($\check{x}$) **do**
      $x_{i,j} \leftarrow \check{x}_{i,j}$
      if $255 - \vartheta \geq x_{i,j} \geq 255 - 2\vartheta + 1$ **then**
        if $v_t = 1$ **then**   ▷ overflow
          $x_{i,j} \leftarrow \check{x}_{i,j} + \vartheta$
        $t = t + 1$
      else if $0 + \vartheta \leq x_{i,j} \leq 0 + 2\vartheta - 1$ **then**
        if $v_t = 1$ **then**   ▷ overflow
          $x_{i,j} \leftarrow \check{x}_{i,j} - \vartheta$
        $t = t + 1;$

repeated once more to embed another part of the payload by swapping the roles between the context and the query.

At the decoder side, we carry out message extraction and image restoration in a *last-in-first-out* manner. Similar to the encoding process, the decoding process begins by splitting pixels of the stego image into the context and the query, predicting the query from the context, and calculating the prediction errors. A partial payload is extracted and the prediction errors are de-modulated back to the original state through inverse operations. Steganographic distortion is removed by adding the restored prediction errors to the predicted values. The decoding process can likewise be repeated once more by swapping roles between the context and query. Finally, the restored image is post-processed with the extracted auxiliary information to undo the minute modifications caused by the overflow prevention measure.

To summarise, the algorithmic steps are enumerated as follows. For the encoding phase:

1) $\{\check{x}, v\} = \text{scale}_{\Downarrow}(x)$;
2) $p = \text{concat}(v, m)$;
3) $\{x_{\mathbb{B}}, x_{\mathbb{W}}\} = \text{split}(\check{x})$;
4) $y_{\mathbb{W}} = \text{net}_1(x_{\mathbb{B}}, x_{\mathcal{Q}} | \mathcal{Q} = \mathbb{W})$;
5) $x'_{\mathbb{W}} = \text{encode}(x_{\mathbb{W}}, y_{\mathbb{W}}, p_1)$;
6) $y_{\mathbb{B}} = \text{net}_2(x'_{\mathbb{W}}, x_{\mathcal{Q}} | \mathcal{Q} = \mathbb{B})$;
7) $x'_{\mathbb{B}} = \text{encode}(x_{\mathbb{B}}, y_{\mathbb{B}}, p_2)$;
8) $x' = \text{merge}(x'_{\mathbb{B}}, x'_{\mathbb{W}})$.

For the decoding phase:

1) $(x'_{\mathbb{B}}, x'_{\mathbb{W}}) = \text{split}(x')$;
2) $y_{\mathbb{B}} = \text{net}_2(x'_{\mathbb{W}}, x_{\mathcal{Q}} | \mathcal{Q} = \mathbb{B})$;
3) $\{x_{\mathbb{B}}, p_2\} = \text{decode}(x'_{\mathbb{B}}, y_{\mathbb{B}})$;
4) $y_{\mathbb{W}} = \text{net}_1(x_{\mathbb{B}}, x_{\mathcal{Q}} | \mathcal{Q} = \mathbb{W})$;
5) $\{x_{\mathbb{W}}, p_1\} = \text{decode}(x'_{\mathbb{W}}, y_{\mathbb{W}})$;
6) $\check{x} = \text{merge}(x_{\mathbb{B}}, x_{\mathbb{W}})$;
7) $\{v, m\} = \text{deconcat}(p)$;
8) $x = \text{scale}_{\Uparrow}(\check{x}, v)$.

All images are assumed to be 8-bit greyscale throughout this study. Let us denote by ϑ a threshold for the stego channel such that

$$\text{stego channel: } \{\varepsilon \mid \text{abs}(\varepsilon) < \vartheta\}, \tag{1}$$

where abs denotes the absolute value. In other words, we use the parameter ϑ to determine for which prediction error values the payload embedding process can take place. The following presents the algorithmic details surrounding the notion of the stego channel.

TABLE 2. Code charts for overflow prevention w.r.t. different settings of ϑ.

(a) $\vartheta = 1$

x	0	255
$\check{x}$	1	254

(b) $\vartheta = 2$

x	0	1	254	255
$\check{x}$	2	3	252	253

(c) $\vartheta = 3$

x	0	1	2	253	254	255
$\check{x}$	3	4	5	250	251	252

B. OVERFLOW HANDLING

Both encoding and decoding are carried out in the residual domain. Addition and subtraction on residual values are equivalent to the same arithmetic operations on pixel values (i.e. a pro rata increment or decrement in pixel intensity). Given that computations are defined in a Galois finite field, such arithmetic operations may cause pixel intensity overflow: intensity values that are unexpectedly small or large wrap around the minimum or maximum after manipulations. We pre-process the image to prevent pixels from moving off boundary (or becoming saturated) and mark the locations where off-boundary pixels may occur. To distinguish between

TABLE 3. Code charts for prediction-error modulation w.r.t. different settings of ϑ.

(a) $\vartheta = 1$

ε	-254	$\cdots$	-3	-2	-1	0			$+1$	$+2$	$+3$	$\cdots$	$+254$
p	$\varnothing$	$\cdots$	$\varnothing$	$\varnothing$	$\varnothing$	10_2	0_2	11_2	$\varnothing$	$\varnothing$	$\varnothing$	$\cdots$	$\varnothing$
ε'	-255	$\cdots$	-4	-3	-2	-1	0	$+1$	$+2$	$+3$	$+4$	$\cdots$	$+255$

(b) $\vartheta = 2$

ε	-253	$\cdots$	-3	-2	-1		0			$+1$		$+2$	$+3$	$\cdots$	$+253$
p	$\varnothing$	$\cdots$	$\varnothing$	$\varnothing$	1_2	0_2	10_2	0_2	11_2	0_2	1_2	$\varnothing$	$\varnothing$	$\cdots$	$\varnothing$
ε'	-255	$\cdots$	-5	-4	-3	-2	-1	0	$+1$	$+2$	$+3$	$+4$	$+5$	$\cdots$	$+255$

(c) $\vartheta = 3$

ε	-252	$\cdots$	-3	-2		-1		0			$+1$		$+2$		$+3$	$\cdots$	$+252$
p	$\varnothing$	$\cdots$	$\varnothing$	1_2	0_2	1_2	0_2	10_2	0_2	11_2	0_2	1_2	0_2	1_2	$\varnothing$	$\cdots$	$\varnothing$
ε'	-255	$\cdots$	-6	-5	-4	-3	-2	-1	0	$+1$	$+2$	$+3$	$+4$	$+5$	$+6$	$\cdots$	$+255$

Algorithm 3 Modulation

Input: $\varepsilon, p, t, \vartheta$
Output: ε', t
  ▷ modulation
if $\varepsilon = 0$ **then**
    if $p_t = 0$ **then**   ▷ embed 1 bit
        $\varepsilon' \leftarrow 0$
        $t \leftarrow t + 1$
    else if $p_t = 1$ and $p_{t+1} = 0$ **then**   ▷ embed 2 bits
        $\varepsilon' \leftarrow -1$
        $t \leftarrow t + 2$
    else if $p_t = 1$ and $p_{t+1} = 1$ **then**   ▷ embed 2 bits
        $\varepsilon' \leftarrow +1$
        $t \leftarrow t + 2$
else if $0 < \mathrm{abs}(\varepsilon) < \vartheta$ **then**   ▷ embed 1 bit
    if $p_t = 0$ **then**
        $\varepsilon' \leftarrow 2\varepsilon$
    else
        $\varepsilon' \leftarrow 2\varepsilon + \mathrm{sgn}(\varepsilon) \cdot 1$
    $t \leftarrow t + 1$
else   ▷ non-embeddable
    $\varepsilon' \leftarrow \varepsilon + \mathrm{sgn}(\varepsilon) \cdot \vartheta$
    $t \leftarrow t$

Algorithm 4 De-Modulation

Input: $\varepsilon', p, t, \vartheta$
Output: ε, p, t
  ▷ de-modulation
if $0 \leq \mathrm{abs}(\varepsilon') \leq 1$ **then**
    $\varepsilon \leftarrow 0$
    if $\varepsilon' = 0$ **then**   ▷ extract 1 bit
        $p \leftarrow \mathrm{concat}(p, 0)$
        $t \leftarrow t + 1$
    else if $\varepsilon' = -1$ **then**   ▷ extract 2 bits
        $p \leftarrow \mathrm{concat}(p, 1, 0)$
        $t \leftarrow t + 2$
    else if $\varepsilon' = +1$ **then**   ▷ extract 2 bits
        $p \leftarrow \mathrm{concat}(p, 1, 1)$
        $t \leftarrow t + 2$
else if $1 < \mathrm{abs}(\varepsilon') < 2\vartheta$ **then**   ▷ extract 1 bit
    $\varepsilon \leftarrow \mathrm{floor}(\mathrm{abs}(\varepsilon')/2) \cdot \mathrm{sgn}(\varepsilon')$
    $p \leftarrow \mathrm{concat}(p, \mathrm{mod}(\mathrm{abs}(\varepsilon'), 2))$
    $t \leftarrow t + 1$
else   ▷ non-extractable
    $\varepsilon \leftarrow \varepsilon' - \mathrm{sgn}(\varepsilon') \cdot \vartheta$
    $p \leftarrow p$
    $t \leftarrow t$

---

the processed pixels $\check{x}$ and unprocessed pixels x with the same value, we flag 1 (true) for the former and 0 (false) for the latter in an overflow-status register, that is,

$$\check{x}_{i,j} = \begin{cases} x_{i,j} + \vartheta & \text{if } x_{i,j} \in \mathbb{L}_1, \\ x_{i,j} - \vartheta & \text{if } x_{i,j} \in \mathbb{U}_1, \\ x_{i,j} & \text{otherwise,} \end{cases} \quad (2)$$

and

$$v = \begin{cases} \mathrm{concat}(v, 0) & \text{if } x_{i,j} \in \mathbb{L}_0 \cup \mathbb{U}_0, \\ \mathrm{concat}(v, 1) & \text{if } x_{i,j} \in \mathbb{L}_1 \cup \mathbb{U}_1, \end{cases} \quad (3)$$

where

$$\mathbb{L}_0 = [0 + \vartheta, 0 + 2\vartheta - 1],$$

$$\mathbb{L}_1 = [0, 0 + \vartheta - 1],$$
$$\mathbb{U}_0 = [255 - 2\vartheta + 1, 255 - \vartheta],$$
$$\mathbb{U}_1 = [255 - \vartheta + 1, 255]. \quad (4)$$

The parameter ϑ is associated with the modulation step in the message embedding process. A greater value of ϑ spans a wider range of pixel values that may be modified beyond the boundary and thus having a higher probability of increasing the size of the register. The overflow-status register is an overhead of reversibility. It has to be concatenated with the message and embedded as a part of the payload. As a consequence, its size has to be deducted from the overall payload when assessing steganographic capacity. See Algorithms 1

and 2 for the pseudo-codes. The overflow handling process is codified in Table 2.

C. ENCODING/DECODING

Let $y_{i,j}$ denote a prediction of the pixel at location (i, j). For each query pixel, we compute its prediction error by

$$\varepsilon = \check{x}_{i,j} - y_{i,j}. \tag{5}$$

For the prediction errors determined as the stego channel, we embed a bit of payload information by

$$\varepsilon' = \varepsilon + \text{sgn}(\varepsilon) \cdot p_t, \tag{6}$$

where sgn extracts the sign of the prediction error (either positive or negative) and p_t is the current payload bit. For the rest of the errors, we shift them by ϑ to prevent error values from overlapping (i.e. indistinguishable from the errors determined as stego channel); that is,

$$\varepsilon' = \varepsilon + \text{sgn}(\varepsilon) \cdot \vartheta. \tag{7}$$

A modulated error is then added to the predicted value, resulting in a stego pixel

$$x'_{i,j} = y_{i,j} + \varepsilon'. \tag{8}$$

Let us take the case in which $\vartheta = 2$ for example. When $\varepsilon = 0$, we can map it to one of the three states, $\varepsilon' = 0$ or ± 1, to encode a ternary digit (a trit of information). When $\varepsilon = \pm 1$, we shift it to ± 2 to avoid ambiguity and map it to one of the two states, $\varepsilon' = \pm 2$ or ± 3 (disregarding the sign), to represent one bit of information. For all other ε such that $\varepsilon \notin (-\vartheta, +\vartheta)$, we shift each of them by ϑ in either a positive or negative direction, depending on the sign. However, converting the payload between binary (base 2) and ternary (base 3) numeral systems on the fly can be problematic. To circumvent this issue, we can map the errors with value 0 to the digit 0 if the observed payload bit p_t is 0_2; otherwise, we read the next payload bit p_{t+1} and map the errors to the digit -1 if $[p_t, p_{t+1}] = [1_2, 0_2]$ and to the digit $+1$ if $[p_t, p_{t+1}] = [1_2, 1_2]$; that is,

$$\{\varepsilon' \mid \varepsilon = 0\} = \begin{cases} 0 & \text{if } p_t = 0_2, \\ -1 & \text{if } [p_t, p_{t+1}] = [1_2, 0_2], \\ +1 & \text{if } [p_t, p_{t+1}] = [1_2, 1_2]. \end{cases} \tag{9}$$

Theoretically, the mapping permits one trit or $\log_2 3 \approx 1.585$ bits of information to be embedded. In practice, the compromised solution embeds one or two bits with a probability of 0.5 each, thereby being capable of embedding 1.5 bits on average for $\varepsilon = 0$. Decoding is simply an inverse mapping. It begins by predicting the query pixel intensities and computing the prediction errors. Each error is de-modulated according to its magnitude. For errors between ± 1, we map them back to 0 and extract the corresponding payload bits. For errors whose magnitude is in $(1, 2\vartheta)$ regardless of the sign, we de-modulate them by the floor division of the magnitude by 2 and extract the payload bits by the

remainder of the division. For the rest errors, we shift them back by ϑ. The original pixel intensity is recovered by adding the de-modulated error to the predicted value.

(a) Cover **(b)** Stego ($\vartheta = 1$) **(c)** Stego ($\vartheta = 2$) **(d)** Stego ($\vartheta = 3$)

(e) NA **(f)** 48.3806 dB **(g)** 43.6148 dB **(h)** 41.4330 dB

(i) NA **(j)** 0.3386 bpp **(k)** 0.6423 bpp **(l)** 0.8296 bpp
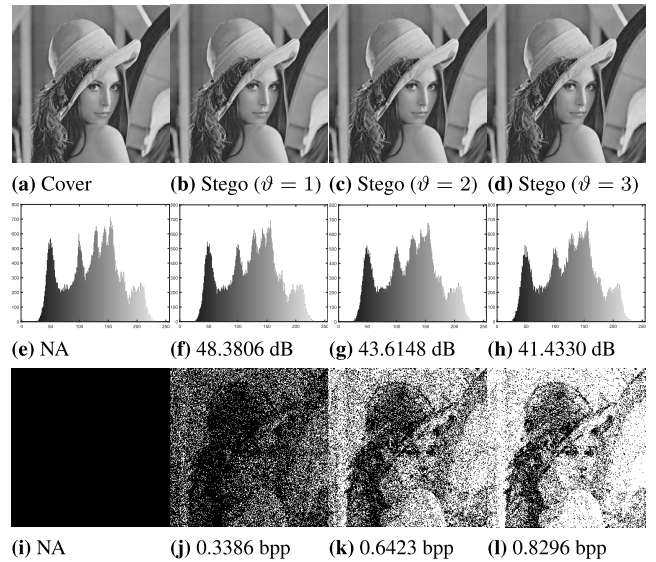
FIGURE 2. Stego images, their tonal distributions, and locations of carrier pixels (coloured in white). Numerical data expresses PSNR and embedding rate.

According to the *Law of Error* [42], we can hypothesise that the frequency of an error approximates an exponential function of its magnitude and therefore follows a zero-mean Laplacian distribution. This symmetrical modulation pattern is derived from the fact that prediction errors are expected to centre around 0. Therefore, it is advisable to designate errors with a small absolute magnitude (of high frequency) as the stego channel. A rise in the width of the stego channel represents an increased steganographic capacity. See Algorithms 3 and 4 for the pseudo-codes. The prediction-error modulation process is codified in Table 3. Figure 2 displays stego images, their tonal distributions, and locations of carrier pixels w.r.t. different settings of stego-channel parameter ϑ. Results are generated by spreading a pseudo-random binary message over the whole image at the maximal steganographic capacity. It can be observed that the quality degradation is virtually imperceptible, the disturbance to the tonal distribution is subtle, and the pixels selected for carrying the payload are mostly clustered in smooth areas. We would like to note that this coding method is simple, readily scalable and computationally efficient but not necessarily optimal. The main theme of this study is the analytics module, which is independent of the coding module. Any coding method that reflects more accurately the statistical distribution of residuals can be applied without conflicting with the findings of this study. For the subject of mathematical optimisation of reversible steganographic coding, the interested reader is referred to the research study [43].

IV. PREDICTIVE ANALYTICS

In this section, we take a closer look at the practice of training neural networks for predictive analytics in reversible steganography. We investigate different initialisation strategies for configuring the inputs of neural networks, different training strategies for fitting neural networks with dual-layer prediction, and different neural network architectures suitable for context-aware pixel intensity prediction.

A. CONTEXT–QUERY SPLITTING

A convenient way to define pixel connectivity is to consider the von Neumann neighbourhood, that is, four adjacent pixels connected horizontally and vertically. If we sample the context and query pixels uniformly in such a way that each query pixel has 4 connected context pixels, we end up forming a pattern analogous to a chequerboard, as illustrated in Figure 3. We can define the black set $\mathbb{B}$ as the context and the white set $\mathbb{W}$ as the query (or the other way round):

$$(i, j) \in \begin{cases} \mathbb{B} := \mathcal{C} & \text{if } i + j \text{ is odd,} \\ \mathbb{W} := \mathcal{Q} & \text{if } i + j \text{ is even.} \end{cases} \quad (10)$$

A naïve way to estimate the intensity of the central pixel is to calculate the mean of locally connected pixels. This heuristic predictive model, referred to as local-mean interpolation (LMI) [44], is based on the smoothness prior, a generic contextual assumption on real-world photographs. The recent studies have shown that neural network models (e.g. MS-CNN [38] and MemNet [39]) can be applied to improve predictive accuracy. Because predictive accuracy is instrumental in steganographic rate-distortion performance, the factors that could have the impacts upon predictive accuracy are primary concerns of this study.
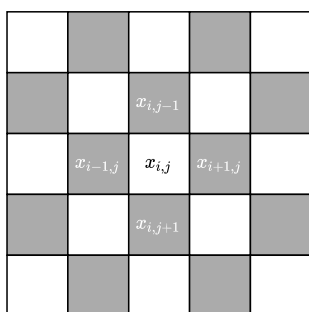
FIGURE 3. Illustration of chequered pattern for context–query splitting.

B. INITIALISATION STRATEGIES

The first step of the encoding phase is to split a cover image into $x_{\mathbb{B}}$ and $x_{\mathbb{W}}$. While the values of the query pixels are supposed to be predicted, the initial values of query pixels $x_{\mathcal{Q}}$ have to be determined and cannot be set as null since the input to the applied neural networks must be a complete image. To this end, we consider two simple initialisation strategies: zero initialisation and local-mean initialisation.

1) ZERO INITIALISATION

For zero initialisation, we set the initial value for each query pixel explicitly to 0. This strategy is rather arbitrary but computationally efficient when compared with other initialisation strategies. Apart from this, it involves little subjective knowledge, preliminary analysis, or preconception about the intensities. In other words, it involves minimal human interference in the machine learning process. We may perceive prediction with zero initialisation as a special type of image super-resolution problem in which pixels are downsampled according to a special chequered pattern. The query pixels set to zero are viewed as missing data.

2) LOCAL-MEAN INITIALISATION

For local-mean initialisation, we assign the mean intensity of 4 connected neighbours as the initial value of each query pixel. This strategy involves the computation of a local mean, but it is often the case that pre-estimation could help to accelerate the training process. We can test whether learning-based models tend to perform better and converge faster when such approximate values are pre-calculated. From another perspective, local-mean initialisation formulates the prediction problem as a image denoising problem by viewing the query pixels as noisy data.

C. TRAINING STRATEGIES

Dual-layer prediction entails the problem of distributional shift because steganographic distortion appears after the first-layer embedding. In the first layer, the intensities of query pixels are predicted by a neural network model and then modified with the prediction-error modulation algorithm. In the second layer, the roles of context and query are switched. As a result, the context set consists of distorted pixels. This can also be viewed as uncertainty propagation in the sense that errors in the previous layer propagate to the next layer, impairing the predictive performance. Formally, we aim to minimise the loss of two test sets:

$$\begin{aligned} &\text{test}_1 : \mathcal{L}(\text{net}(x_{\mathbb{B}}, x_{\mathcal{Q}}), x_{\mathbb{W}}), \\ &\text{test}_2 : \mathcal{L}(\text{net}(x'_{\mathbb{W}}, x_{\mathcal{Q}}), x_{\mathbb{B}}). \end{aligned} \quad (11)$$

To this end, we explore three training strategies: universal training, independent training, and causal training. The training strategies differ by the configurations of the training set, as illustrated in Figure 4.

1) UNIVERSAL TRAINING

The universal training is a cost-efficient way to manage dual-layer prediction by training a single model for performing prediction in both layers. The motivation is that the context/query switch between the first and second layers can be perceived as a simple geometric translation and convolutional neural networks (CNNs) are considered to be translation-invariant. For that reason, a single model may be sufficient and can generalise well for tackling dual-layer prediction. According to the chequered pattern, the context

(or query) coordinates of the first layer, when shifted by one step (either horizontally or vertically), becomes the context (or query) coordinates of the second layer. Inspired by the translation-invariance property of classic CNN models, we conjecture that a model trained on the first set can be deployed directly to make inferences on the second test set. In other words, for both test sets, we train a universal model by

$$\text{net}_{\text{uni}} : \underset{\text{net}}{\arg\min} \, \mathcal{L}(\text{net}(x_{\mathbb{B}}, x_{\mathcal{Q}}), x_{\mathbb{W}}). \tag{12}$$

2) INDEPENDENT TRAINING

The independent training is to train two models in parallel for respective layers. For inference on the second test set, we structure the training set without taking account of steganographic distortion. This strategy can serve to check whether a single translation-invariant model is adequate. If the conjecture is valid, both would yield a similar result. Although the practice of training two models indicates an extra computational cost, this strategy permits parallel training because the unmodified images rather than stego images are used as the inputs for training the second-layer predictor, that is,

$$\text{net}_{\text{ind}}^1 : \underset{\text{net}}{\arg\min} \, \mathcal{L}(\text{net}(x_{\mathbb{B}}, x_{\mathcal{Q}}), x_{\mathbb{W}}),$$
$$\text{net}_{\text{ind}}^2 : \underset{\text{net}}{\arg\min} \, \mathcal{L}(\text{net}(x_{\mathbb{W}}, x_{\mathcal{Q}}), x_{\mathbb{B}}). \tag{13}$$

3) CAUSAL TRAINING

The causal training is to train the models in a consecutive manner such that the succeeding model is trained after the data has been modified with the preceding model. That is, a predictive model is trained and then deployed for embedding information into the training data for another model. The problem of distributional shift stems from the discrepancy between the training and deployment environments (i.e. distributions of the training and test sets). Reducing the deployment loss requires the distribution of the training set to be as close as possible to that of the test set. A potential way to remedy the problem is to inject steganographic distortion to the second-layer training set. At the first glance, it seems not surprising that the distributional shift can be mitigated by introducing steganographic distortion to the training of the second layer. However, steganographic distortion varies with the embedded message and is quite random (rather than being a fixed noise pattern). It implies that such distortion can not be simply filtered out by the models. Nevertheless, we hypothesis that neural networks can learn to predict from much more stable and reliable image representations if such distortion is presented during training. Implementing a steganographic algorithm incurs an extra computational cost and the succeeding model can only be trained once the training of the preceding model has been completed. Two models are causally connected as the prediction of the first model contributes to the prediction of the second model.

Specifically, two models are trained by

$$\text{net}_{\text{cos}}^1 : \underset{\text{net}}{\arg\min} \, \mathcal{L}(\text{net}(x_{\mathbb{B}}, x_{\mathcal{Q}}), x_{\mathbb{W}}),$$
$$\text{net}_{\text{cos}}^2 : \underset{\text{net}}{\arg\min} \, \mathcal{L}(\text{net}(x'_{\mathbb{W}}, x_{\mathcal{Q}}), x_{\mathbb{B}}). \tag{14}$$
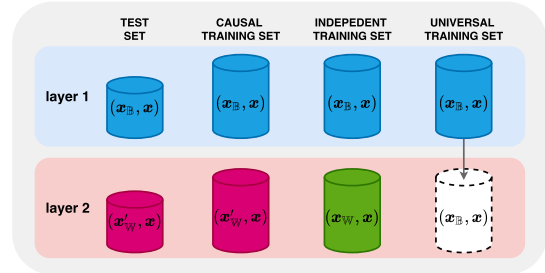
FIGURE 4. Training sets of causal, indepedent and universal strategies for dual-layer predictive analytics.

D. NEURAL NETWORKS

An image whose query pixels are initialised to either zero or local mean can be viewed as a noisy and low-resolution version of the original image. The goal of context-aware pixel-intensity prediction can therefore be considered as refining the observed image into a clean and high-resolution one. Based on this perception, we can adopt advanced deep-learning models originally devised for image denoising and image super-resolution. A neural network is essentially a non-linear function that learns to minimise a pre-defined loss function by transforming the input into useful features or representations in a latent space. For the task of pixel-intensity prediction, we suggest training the models to minimise the ℓ_1 norm or mean absolute error (MAE). The reason for choosing the ℓ_1 norm over the ℓ_2 norm is that the latter tends to produce overly smoothed outputs in image enhancement tasks [45], [46], [47]. The ℓ_2 norm, or mean squared error (MSE), encourages producing an average of plausible solutions and is prone to being affected by outliers. There are other preferable loss functions in the field of low-level computer vision. An example is the Euclidean distance between the high-level feature maps extracted by a 19-layer VGG neural network [48] pre-trained on the ImageNet (a database for large-scale visual recognition) [49]. Another example is the adversarial loss which uses a discriminator to estimate the likelihood that a generated instance is real [50]. Nevertheless, we do not anticipate these ad hoc loss functions improving steganographic performance because prediction-error modulation relies mainly on pixel-wise distance. Apart from the choice of loss functions, when applying off-the-shelf models, upsampling and transposed convolutional layers should be replaced with regular convolutional layers because the sizes of the input and output images are the same in pixel-intensity prediction. While the neural network architectures vary from one to another and it is difficult to foresee their performance in different tasks (due to the 'black box' nature of neural networks), we attach relative importance to shortcut connection

because it mitigates the performance degradation problem when increasing network depth [51] and allows models to learn explicitly the minute differences between the input and output images [52]. To ensure reproducibility, the architectural details of the MS-CNN, MemNet and RDN used for empirical evaluation are illustrated in Figure 5 and specified as follows.
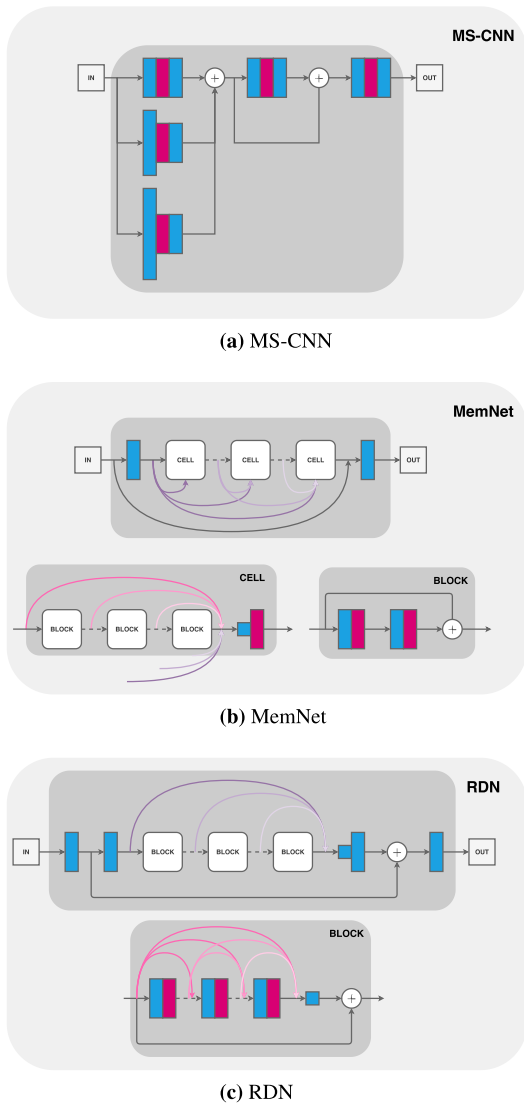

(a) MS-CNN

(b) MemNet

(c) RDN

FIGURE 5. Architectural details of neural networks for context-aware pixel-intensity prediction. Blue boxes, red boxes and plus sign represent covlutional layers, activation functions and element-wise sum respectively.

1) MS-CNN
The MS-CNN extracts multi-scale image features with convolutional kernels of different sizes parallel to one another, to overcome the problem of restricted receptive fields in traditional linear predictors. The multi-scale features are aggregated in two successive convolutional layers to make a final prediction. In the implementation, the multi-scale kernel sizes

are configured to 3×3, 5×5, and 7×7 with the number of channels set to 32.

2) MemNet
The MemNet consists of interconnected memory cells, each comprising a recurrent unit and a gate unit. The recurrent connectivity substantially reduces the number of trainable parameters, enabling the formation of a lightweight model for storage. The gating mechanism regulates important latent states or persistent memories, thereby mitigating the vanishing gradient problem often encountered when training deep neural networks. The recurrent unit comprises a series of residual blocks with shared weights, whereas the gating mechanism is a convolutional layer with a kernel size of 1×1. For each memory cell, the outputs from the tightly structured residual blocks (short-term memories) along with the outputs from previous cells pass through a gate unit to attain a persistent state (long-term memory). In the implementation, the number of memory cells and the number of residual blocks in each cell are both configured to 5 with the kernel size set to 3×3 and the number of channels set to 64.

3) RDN
The RDN combines residual connections and dense connections to learn hierarchical image representations. The model exploits hierarchical representations to the fullest by fusing features at both the global and local levels. At the local level, convolutional layers are densely connected, forming a residual dense block with a skip connection between the input and output. A convolutional layer with a kernel size of 1×1 is used to fuse the local feature maps. At the global level, the outputs of multiple blocks are, again, blended via a convolutional layer with kernel size 1, followed by a skip connection from a shallow layer to a deep layer. In the implementation, the number of residual dense blocks is configured to 3 and the number of convolutional layers in each block is configured to 5 with the kernel size set to 3×3 and the number of channels set to 64.

V. EXPERIMENTS
In this section, we report and discuss experimental results w.r.t. different predictive models. We begin by discussing the choice of benchmarks and describing the general set-up of experiments. Then, we evaluate the impacts of different initialisation and training strategies on predictive accuracy. To further evaluate the performance of different models, we analyse the distribution of prediction errors. Finally, we verify a direct correlation between predictive accuracy and steganographic performance by examining the steganographic rate-distortion curves.

A. BENCHMARKS
The main theme of this study is predictive analytics. For a fair comparison, we evaluate the performance of different predictive models based on the same steganographic algorithm (i.e. prediction-error modulation). We do not compare
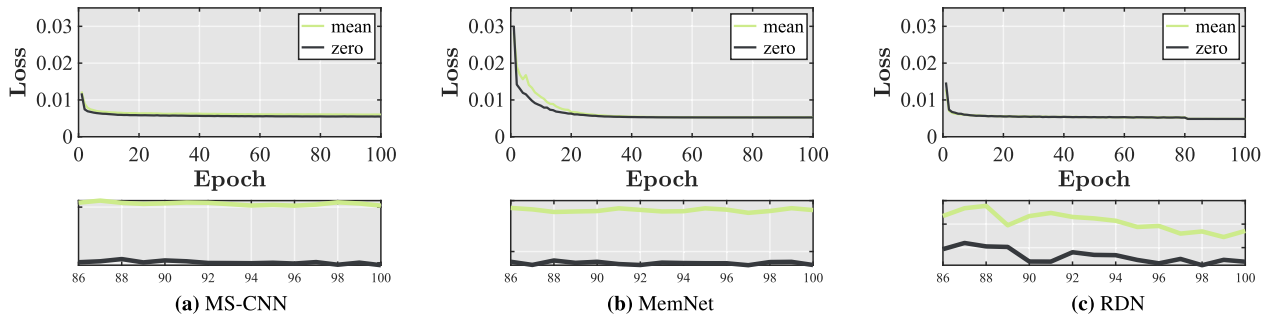
FIGURE 6. Learning curves for zero and local-mean initialisation strategies over 100 epochs. Top plot shows training loss over 100 epochs. Bottom plot shows zoomed-in section from 86th to 100th epoch.
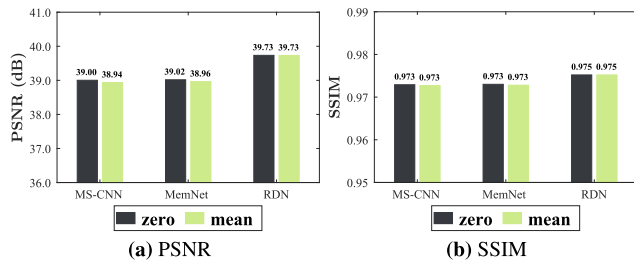

FIGURE 7. Visual quality assessment for first-layer prediction with zero and local-mean initialisation.
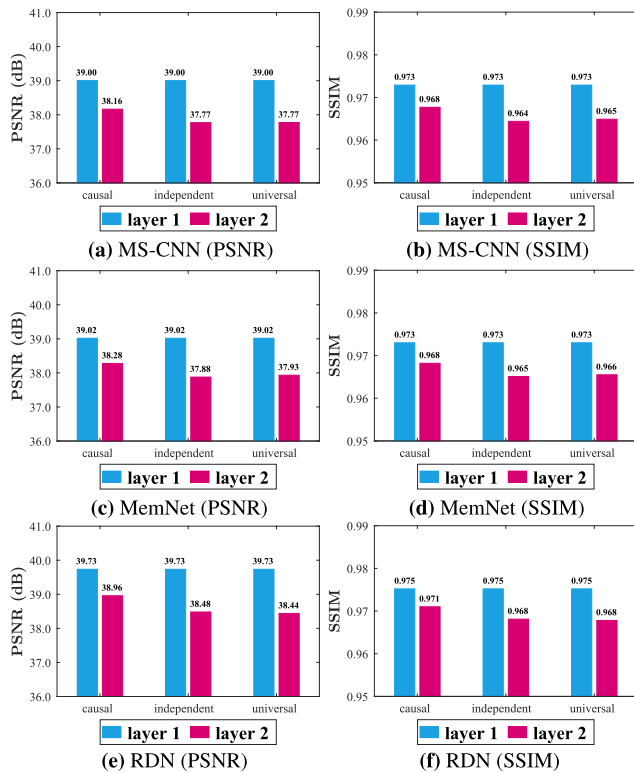

FIGURE 8. Visual quality assessment for dual-layer prediction with causal, independent and universal training.

standpoint. In general, the end-to-end framework cannot reliably satisfy the requirement for perfect reversibility given the limit of current deep-learning algorithms. For the predictive analytics module, we select LMI as a benchmark model [44] based on heuristics and both MS-CNN [38] and MemNet [39] as state-of-the-art models based on deep learning. All the selected models are specifically designed and applied for the prediction task with the same context–query arrangement (i.e. the chequered pattern). The mentioned deep learning models have also been shown to outperform other traditional methods for either same or different context–query arrangement, including nearest-neighbour interpolation [53], median-edge detector [54], gradient-adjusted predictor [55] and bilinear interpolation [56]. In summary, the coding algorithm and the context–query splitting pattern are held constant (as control variables) throughout the course of the investigation to prevent the interference in the experimental results. We select LMI, MS-CNN and MemNet as representative benchmarks and carry out a comparative study with the advanced RDN model.

B. EXPERIMENTAL SETUP

The neural network models are trained and tested on the BOSSbase dataset [57], which originated from an academic competition for digital steganography. It comprises a collection of $10,000$ greyscale photographs covering a wide variety of subjects and scenes. To fit the models over a broad range of hardware options with reasonable training time, all the images are resized to a resolution of 256×256 pixels by using the Lanczos resampling algorithm [58]. The training and test sets are randomly sampled at a ratio of 80/20. All the neural network models are trained to minimise the ℓ_1 loss unless specified otherwise. Further inferences and evaluations are made on a set of standard test images from the volume 'miscellaneous' in the USC-SIPI dataset [59]. To investigate predictive performance on challenging samples, we also take Brodatz texture images from the volume 'textures' in the USC-SIPI dataset [60]. We use peak signal-to-noise ratio (PSNR) (expressed in decibels; dB) [61] and structural similarity (SSIM) [62] for perceptual quality assessment. When evaluating the visual quality of predicted
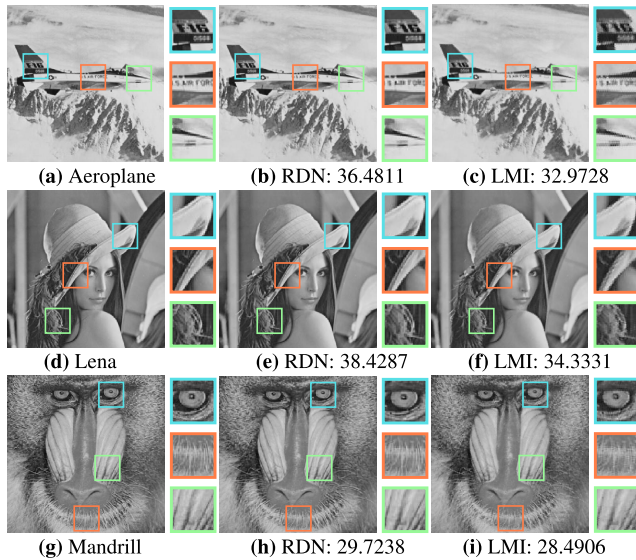
steganographic performance with the end-to-end framework because it approaches the problem from a different

FIGURE 9. Visual comparisons for selected images from USC-SIPI 'miscellaneous' volume. Numerical data expresses PSNR (in dB).

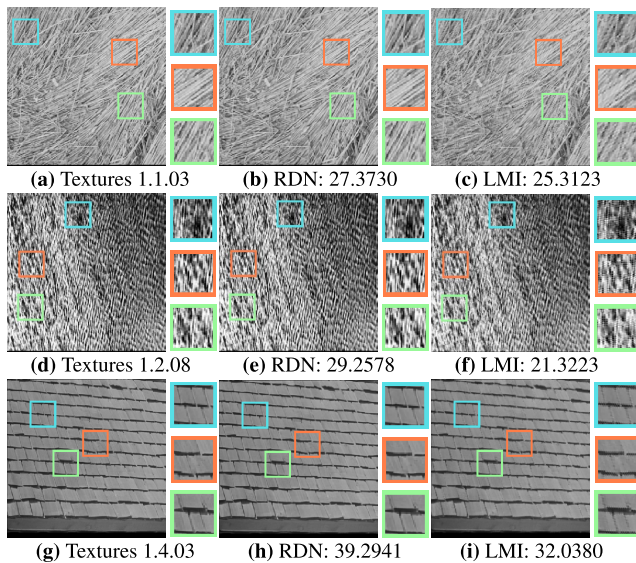(a) Aeroplane   (b) RDN: 36.4811   (c) LMI: 32.9728

(d) Lena   (e) RDN: 38.4287   (f) LMI: 34.3331

(g) Mandrill   (h) RDN: 29.7238   (i) LMI: 28.4906

FIGURE 10. Visual comparisons for selected images from USC-SIPI 'textures' volume. Numerical data expresses PSNR (in dB).

(a) Textures 1.1.03   (b) RDN: 27.3730   (c) LMI: 25.3123

(d) Textures 1.2.08   (e) RDN: 29.2578   (f) LMI: 21.3223

(g) Textures 1.4.03   (h) RDN: 39.2941   (i) LMI: 32.0380

images, we take the whole image into account because the bias in the context pixels is almost negligible. The context pixels in each predicted image are almost always the same as those in the original image. We use embedding rate (expressed in bits per pixel; bpp) for steganographic capacity evaluation.

C. COMPARISON OF INITIALISATION STRATEGIES

We compare the zero and local-mean initialisation strategies by the learning curves and the perceptual qualities. The learning curves are plotted in Figure 6. The curves represent the training loss over 100 epochs for the input/target pairs $(x_{\mathbb{B}}, x)$. It can be observed that while the convergence rate

varies from model to model, zero initialisation reaches a slightly lower loss than the local-mean initialisation in the last epoch. A possible explanation is that although setting the pixel intensity to zero seems to be abrupt and oversimplified, this approach involves minimal human intervention in the machine learning pipeline. By contrast, local-mean initialisation can be viewed as introducing subjective prior knowledge (i.e. the smoothness prior) in the training process. The perceptual qualities of predicted images are measured in Figure 7. Through observing visual quality measurements on different models, we conclude that there is virtually no difference between the impacts of two initialisation strategies upon predictive accuracy. While local-mean initialisation provides a rough estimation in advance, its contribution to predictive accuracy is negligible. Since zero initialisation has a virtue of low computational complexity, we adopt it for the remaining experiments.

D. COMPARISON OF TRAINING STRATEGIES

Predictive accuracies of the universal, independent, and causal training strategies for dual-layer prediction are depicted in Figure 8. The bars show the average PSNR and SSIM scores between the ground-truth and predicted images w.r.t. different training strategies. Steganographic distortion introduced in the first layer would propagate and decrease the predictive accuracy of the second layer. As a result, the second-layer scores are generally lower than the first-layer scores due to the distributional shift. The causal training achieves a comparatively high accuracy for the second-layer prediction, whereas the performance gap between the other two strategies is not distinct. This suggests that the causal training using a training set with a distribution similar to that of the test set can indeed alleviate the distributional shift to some extent. The narrow performance gap between the universal and independent training can be attributed to the translation-invariance property of CNNs. It suggests that training a single model can be as good as training two separate models. Although causal training requires additional computations when constructing the second training set, it is proved to be the most effective among the three training strategies. Hence, we opt for the causal training strategy for the remaining experiments.

E. EVALUATION OF VISUAL QUALITY

Visual comparisons between a heurstic model and a neural network model are shown in Figures 9 and 10. We evaluate the PSNR scores for images predicted from the LMI and RDN. The reported scores are related to the first-layer prediction. A close inspection of zoomed-in views reveals that the RDN is better able to retrieve textural areas with reasonable accuracy in comparison with the naïve LMI, owing to the ability of neural networks to learn rich patterns. Figure 11 compares predictive accuracy of the RDN, MemNet, MS-CNN and LMI on texture images. The results reflect a significant
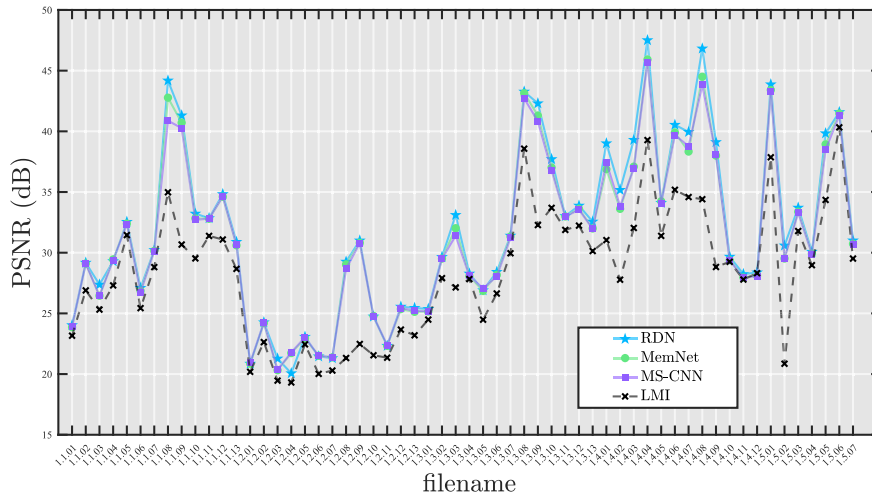
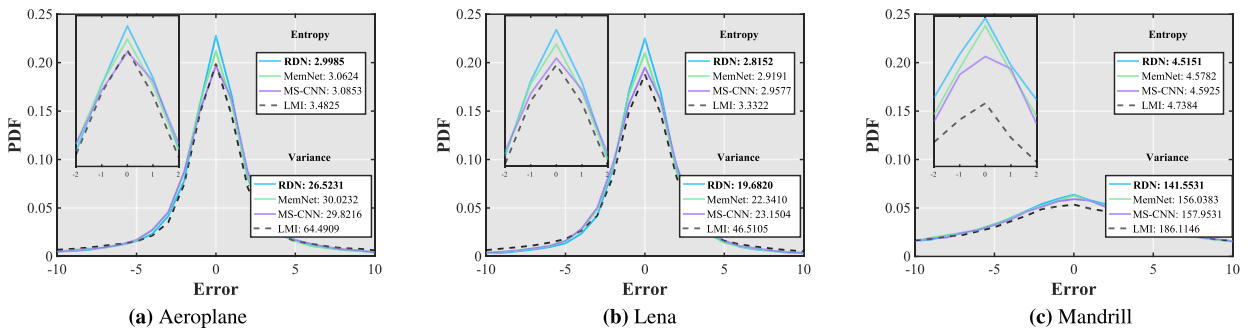**FIGURE 11.** Visual quality assessment for first-layer prediction on Brodatz texture images.

(a) Aeroplane **(b)** Lena **(c)** Mandrill

FIGURE 12. Probability distribution function of errors. Numerical data expresses entropy and variance.

(a) Aeroplane **(b)** Lena **(c)** Mandrill
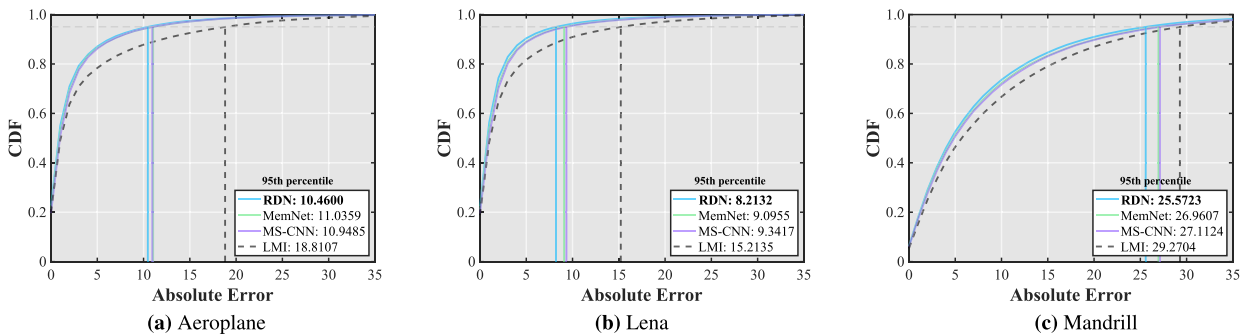
FIGURE 13. Cumulative distribution function of errors. Numerical data expresses 95th percentile.
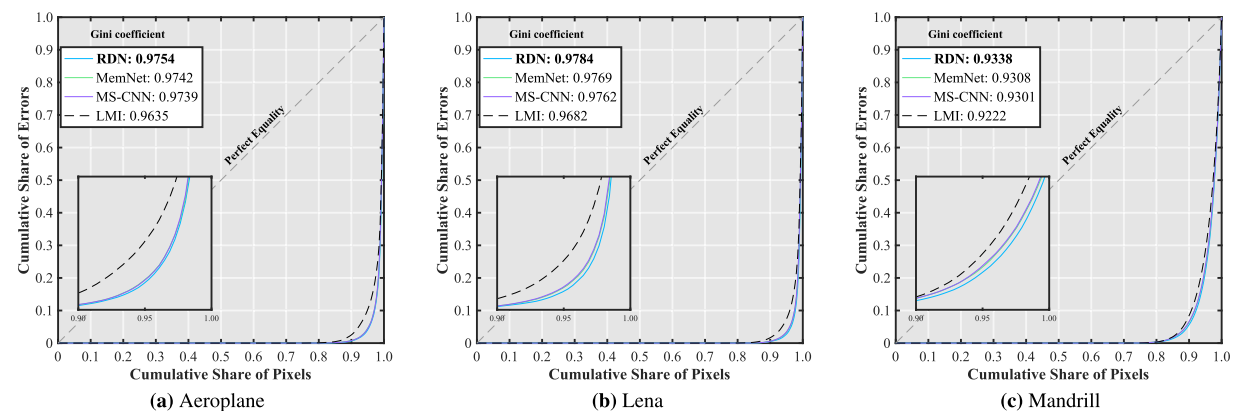

(a) Aeroplane **(b)** Lena **(c)** Mandrill

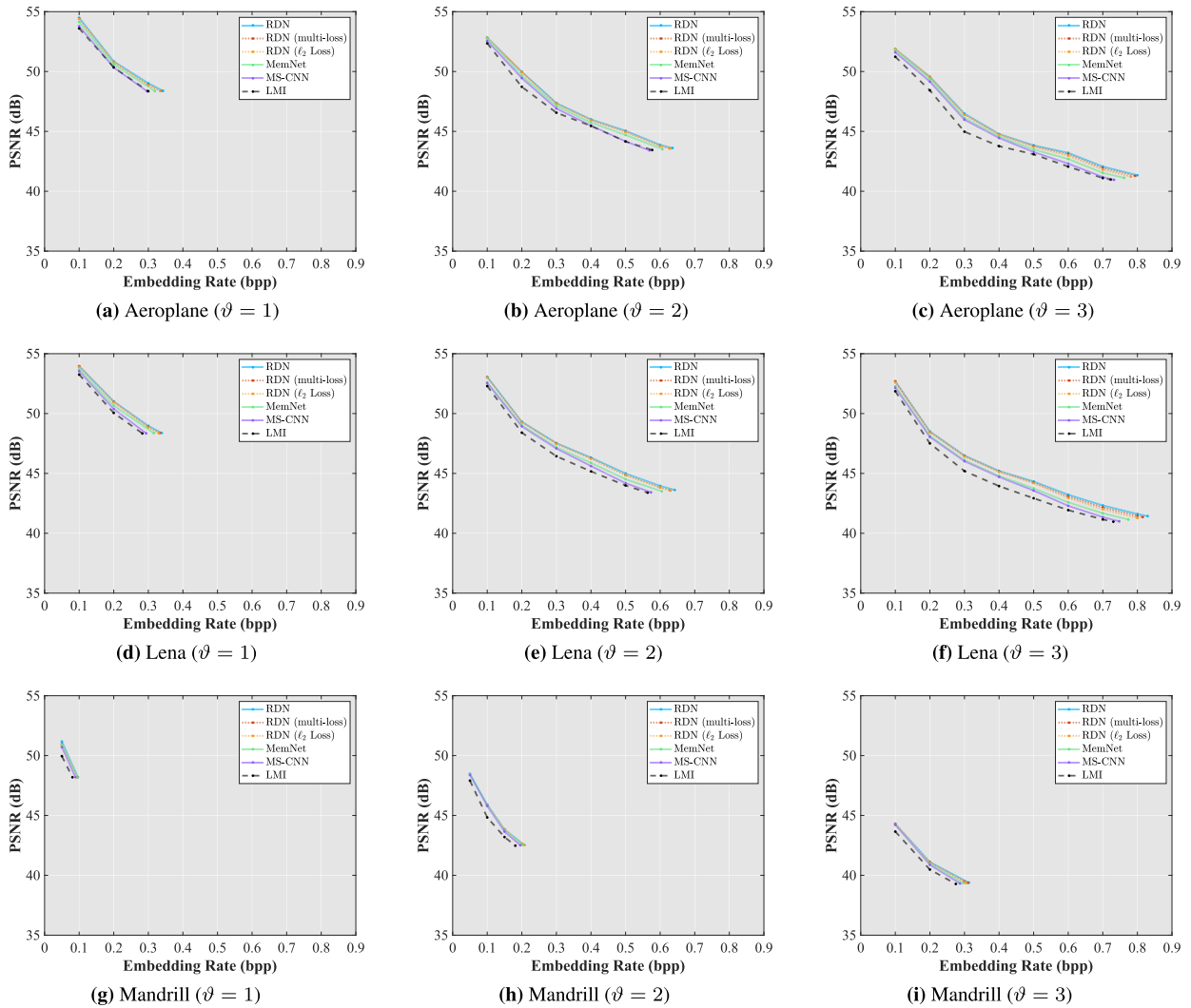**FIGURE 14.** Lorenz curve of errors. Numerical data expresses Gini coefficient.

FIGURE 15. Rate-distortion curves for different settings of stego-channel parameter ϑ.

improvement yielded by the RDN model over other predictive models, confirming the capability of the former to predict textural patterns.

F. EVALUATION OF DISTRIBUTION OF ERRORS

The distribution of prediction errors plays a pivotal role in steganographic performance. Normally, errors would follow a zero-centred Laplacian distribution and a more accurate model results in a more concentrated distribution. According to the coding algorithm, a more concentrated distribution can lead to a better steganographic rate-distortion trade-off. To analyse the error distribution, we examine the probability distribution function (PDF), cumulative distribution function (CDF), and Lorenz curve of errors, as shown in Figures 12, 13, and 14, respectively. We use the entropy, variance, percentile statistics, and Gini coefficient to measure the degree of error concentration [63]. Shannon's entropy is maximised for a uniform distribution and the converse is

equally true: a smaller entropy means a more concentrated distribution [13]. The variance is a measure of the spread of the data around the mean. The 95th percentile indicates the maximum error magnitude below which 95% of errors fall. The Gini coefficient is a measure of statistical dispersion intended to represent the inequality of the error magnitude within an image. A coefficient of 0 expresses perfect equality (dispersion), whereas a coefficient of 1 corresponds to maximal inequality (concentration). For the overall degree of concentration, the RDN ranked in the first tier, followed by the MemNet and the MS-CNN in the second tier, with the LMI ranked last.

G. EVALUATION OF RATE-DISTORTION PERFORMANCE

The steganographic rate-distortion curves are plotted in Figure 15. The models with a higher predictive accuracy indeed achieve a better rate-distortion trade-off. The perceptual quality of stego images is inversely proportional

to the embedding rate since distortions accumulate along the embedding process. The maximum capacity depends on the image content. Images with highly textured details would have lower capacity. A rise of ϑ increases the maximum embedding rate at the cost of compromising the rate-distortion trade-off. Amongst all models, the RDN stands out, achieving state-of-the-art steganographic performance. Apart from the ℓ_1 loss, we investigate the applicability of other common loss functions in the field of image restoration. We implement the ℓ_2 loss and the multi-loss on the RDN model. The latter is comprised of ℓ_1 loss, feature-space loss and adversarial loss. While an extra computational cost is devoted to the implmentation of the multi-loss training, the results suggest that there is marginal, if any, gain in steganographic performance by applying such loss function. This reinforces the argument that the prediction-error modulation algorithm relies primarily on pixel-wise distance.

VI. LIMITATIONS OF THE STUDY

While deep learning has revolutionised the research field of reversible steganography and led to major technological breakthroughs in terms of capacity and imperceptibility, novel application scenarios may be offered if secrecy and robustness are taken into further consideration. The current use of neural networks in the modular framework is confined to predictive analytics. By relaxing the constraint on perfect reversibility, it might be possible to apply the end-to-end framework to automatically learn to create stego objects that are undetectable under steganalysis tools and robust against common multimedia processing operations. Apropos of predictive analytics, an important aspect, apart from predictive accuracy, is predictive uncertainty. If the uncertainty about predictions can be estimated, the rate-distortion performance may be further improved by selecting pixels which are predicted with high confidence.

VII. CONCLUSION

In this work, we have discussed unexplored issues such as the impact of intensity initialisation on predictive accuracy and the influence of distributional shift in dual-layer prediction. Experimental results have revealed that setting pixel intensity to zero, albeit seemingly arbitrary, renders a steadily low loss over several epochs. In addition, it has been found that training models in a causal way can, to some extent, ameliorate the distributional shift in deployment since it minimises the discrepancy in the distributions of training and test sets. The state-of-the-art predictive accuracy and steganographic rate-distortion performance can be achieved by applying advanced pixel-level computer vision models. We envision a promising paradigm shift in reversible steganography ushered in by deep learning and hope that this paper can prove instructive for future research.

REFERENCES

[1] R. J. Anderson and F. A. P. Petitcolas, "On the limits of steganography," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 4, pp. 474–481, May 1998.

[2] J. Fridrich, M. Goljan, P. Lisonek, and D. Soukal, "Writing on wet paper," *IEEE Trans. Signal Process.*, vol. 53, no. 10, pp. 3923–3935, Oct. 2005.

[3] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. Image Process.*, vol. 6, no. 12, pp. 1673–1687, Dec. 1997.

[4] D. Kundur and D. Hatzinakos, "Digital watermarking for telltale tamper proofing and authentication," *Proc. IEEE*, vol. 87, no. 7, pp. 1167–1180, Jul. 1999.

[5] J. Fridrich, M. Goljan, and R. Du, "Invertible authentication," *Proc. SPIE*, vol. 4314, pp. 197–208, Aug. 2001.

[6] C. D. Vleeschouwer, J. F. Delaigle, and B. Macq, "Circular interpretation of bijective transformations in lossless watermarking for media asset management," *IEEE Trans. Multimedia*, vol. 5, no. 1, pp. 97–105, Mar. 2003.

[7] G. Coatrieux, W. Pan, N. Cuppens-Boulahia, F. Cuppens, and C. Roux, "Reversible watermarking based on invariant image classification and dynamic histogram shifting," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 1, pp. 111–120, Jan. 2013.

[8] Y.-Q. Shi, X. Li, X. Zhang, H.-T. Wu, and B. Ma, "Reversible data hiding: Advances in the past two decades," *IEEE Access*, vol. 4, pp. 3210–3237, 2016.

[9] I. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, San Diego, CA, USA, 2015, pp. 1–11.

[10] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "DeepFool: A simple and accurate method to fool deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 2574–2582.

[11] L. Muñoz-González, B. Biggio, A. Demontis, A. Paudice, V. Wongrassamee, E. C. Lupu, and F. Roli, "Towards poisoning of deep learning algorithms with back-gradient optimization," in *Proc. ACM Workshop Artif. Intell. Secur.*, Dallas, TX, USA, Nov. 2017, pp. 27–38.

[12] C.-C. Chang, "Reversible linguistic steganography with Bayesian masked language modeling," *IEEE Trans. Computat. Social Syst.*, early access, Aug. 8, 2022.

[13] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, no. 3, pp. 379–423, Jul. 1948.

[14] X. Li, B. Yang, and T. Zeng, "Efficient reversible watermarking based on adaptive prediction-error expansion and pixel selection," *IEEE Trans. Image Process.*, vol. 20, no. 12, pp. 3524–3533, Dec. 2011.

[15] I.-C. Dragoi and D. Coltuc, "Local-prediction-based difference expansion reversible watermarking," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1779–1790, Apr. 2014.

[16] H. J. Hwang, S. Kim, and H. J. Kim, "Reversible data hiding using least square predictor via the LASSO," *EURASIP J. Image Video Process.*, vol. 2016, no. 1, Dec. 2016, Art. no. 42.

[17] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 2472–2481.

[18] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image restoration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 7, pp. 2480–2495, Jul. 2020.

[19] A. M. Turing, "Computing machinery and intelligence," *Mind*, vol. 59, no. 236, pp. 433–460, 1950.

[20] Y. LeCun, Y. Bengio, and G. E. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, Feb. 2015.

[21] J. Zhu, R. Kaplan, J. Johnson, and F.-F. Li, "HiDDeN: Hiding data with deep networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Munich, Germany, 2018, pp. 682–697.

[22] W. Tang, S. Tan, B. Li, and J. Huang, "Automatic steganographic distortion learning using a generative adversarial network," *IEEE Signal Process. Lett.*, vol. 24, no. 10, pp. 1547–1551, Oct. 2017.

[23] W. Tang, B. Li, S. Tan, M. Barni, and J. Huang, "CNN-based adversarial embedding for image steganography," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 8, pp. 2074–2087, Aug. 2019.

[24] J. Yang, D. Ruan, J. Huang, X. Kang, and Y.-Q. Shi, "An embedding cost learning framework using GAN," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 839–851, 2020.

[25] D. Volkhonskiy, I. Nazarov, and E. Burnaev, "Steganographic generative adversarial networks," in *Proc. Int. Conf. Mach. Vis. (ICMV)*, vol. 11433, Amsterdam, The Netherlands, Jan. 2020, pp. 1–15.

[26] J. Hayes and G. Danezis, "Generating steganographic images via adversarial training," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, Long Beach, CA, USA, 2017, pp. 1951–1960.

[27] D. Hu, L. Wang, W. Jiang, S. Zheng, and B. Li, "A novel image steganography method via deep convolutional generative adversarial networks," *IEEE Access*, vol. 6, pp. 38303–38314, 2018.

[28] X. Luo, R. Zhan, H. Chang, F. Yang, and P. Milanfar, "Distortion agnostic deep watermarking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 13545–13554.

[29] E. Wengrowski and K. Dana, "Light field messaging with deep photographic steganography," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 1515–1524.

[30] H. Fang, D. Chen, F. Wang, Z. Ma, H. Liu, W. Zhou, W. Zhang, and N. H. Yu, "TERA: Screen-to-camera image code with transparency, efficiency, robustness and adaptability," *IEEE Trans. Multimedia*, vol. 24, pp. 955–967, 2021.

[31] M. Tancik, B. Mildenhall, and R. Ng, "StegaStamp: Invisible hyperlinks in physical photographs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 2114–2123.

[32] X. Duan, K. Jia, B. Li, D. Guo, E. Zhang, and C. Qin, "Reversible image steganography scheme based on a U-Net structure," *IEEE Access*, vol. 7, pp. 9314–9323, 2019.

[33] Z. Zhang, G. Fu, F. Di, C. Li, and J. Liu, "Generative reversible data hiding by image-to-image translation via GANs," *Secur. Commun. Netw.*, vol. 2019, Sep. 2019, Art. no. 4932782.

[34] S.-P. Lu, R. Wang, T. Zhong, and P. L. Rosin, "Large-capacity image steganography based on invertible neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 10816–10825.

[35] J. Fridrich, M. Goljan, and R. Du, "Lossless data embedding—New paradigm in digital watermarking," *EURASIP J. Adv. Signal Process.*, vol. 2002, no. 2, pp. 185–196, Dec. 2002.

[36] C.-C. Chang, "Adversarial learning for invertible steganography," *IEEE Access*, vol. 8, pp. 198425–198435, 2020.

[37] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 5967–5976.

[38] R. Hu and S. Xiang, "CNN prediction based reversible data hiding," *IEEE Signal Process. Lett.*, vol. 28, pp. 464–468, 2021.

[39] C.-C. Chang, "Neural reversible steganography with long short-term memory," *Secur. Commun. Netw.*, vol. 2021, Apr. 2021, Art. no. 5580272.

[40] Y. Tai, J. Yang, X. Liu, and C. Xu, "MemNet: A persistent memory network for image restoration," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 4549–4557.

[41] W. Yang, X. Zhang, Y. Tian, W. Wang, J.-H. Xue, and Q. Liao, "Deep learning for single image super-resolution: A brief review," *IEEE Trans. Multimedia*, vol. 21, no. 12, pp. 3106–3121, Dec. 2019.

[42] E. B. Wilson, "First and second laws of error," *J. Amer. Stat. Assoc.*, vol. 18, no. 143, pp. 841–851, Sep. 1923.

[43] C.-C. Chang, "Automation of reversible steganographic coding with non-linear discrete optimisation" *Connect. Sci.* vol. 34, no. 1, pp. 1719–1735, 2022.

[44] V. Sachnev, H. J. Kim, J. Nam, S. Suresh, and Y. Q. Shi, "Reversible watermarking algorithm using sorting and prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 7, pp. 989–999, Jul. 2009.

[45] Z. Wang and A. C. Bovik, "Mean squared error: Love it or leave it? A new look at signal fidelity measures," *IEEE Signal Process. Mag.*, vol. 26, no. 1, pp. 98–117, Jan. 2009.

[46] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for image restoration with neural networks," *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 47–57, Mar. 2017.

[47] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 105–114.

[48] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, San Diego, CA, USA, 2015, pp. 1–14.

[49] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and F.-F. Li, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[50] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, Montreal, QC, Canada, 2014, pp. 2672–2680.

[51] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[52] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.

[53] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 8, pp. 890–896, Aug. 2003.

[54] D. M. Thodi and J. J. Rodriguez, "Expansion embedding techniques for reversible watermarking," *IEEE Trans. Image Process.*, vol. 16, no. 3, pp. 721–730, Mar. 2007.

[55] M. Fallahpour, "Reversible image data hiding based on gradient adjusted prediction," *IEICE Electron. Exp.*, vol. 5, no. 20, pp. 870–876, 2008.

[56] L. Luo, Z. Chen, M. Chen, X. Zeng, and Z. Xiong, "Reversible image watermarking using interpolation technique," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 1, pp. 187–193, Mar. 2010.

[57] P. Bas, T. Filler, and T. Pevný, "'Break our steganographic system': The ins and outs of organizing BOSS," in *Proc. Int. Workshop Inf. Hiding (IH)*, Prague, Czech Republic, 2011, pp. 59–70.

[58] C. E. Duchon, "Lanczos filtering in one and two dimensions," *J. Appl. Meteorol.*, vol. 18, no. 8, pp. 1016–1022, 1979.

[59] A. G. Weber, "The USC-SIPI image database: Version 5," USC Viterbi School Eng., Signal Image Process. Inst., Los Angeles, CA, USA, Tech. Rep. 315, 2006.

[60] P. Brodatz, *Textures: A Photographic Album for Artists and Designers*. New York, NY, USA: Dover, 1966.

[61] Q. Huynh-Thu and M. Ghanbari, "Scope of validity of PSNR in image/video quality assessment," *Electron. Lett.*, vol. 44, no. 13, pp. 800–801, Jun. 2008.

[62] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[63] P. E. Hart, "Entropy and other measures of concentration," *J. Roy. Statist. Soc. A*, vol. 134, no. 1, pp. 73–85, 1971.

CHING-CHUN CHANG received the Ph.D. degree in computer science from the University of Warwick, U.K., in 2019. He participated in a short-term scientific mission supported by European Cooperation in Science and Technology Actions at the Faculty of Computer Science, Otto-von-Guericke-Universität Magdeburg, Germany, in 2016. He was granted the Marie-Curie Fellowship and participated in a research and innovation staff exchange scheme supported by Marie Skłodowska-Curie Actions at the Department of Electrical and Computer Engineering, New Jersey Institute of Technology, USA, in 2017. He was a Visiting Scholar with the School of Computer and Mathematics, Charles Sturt University, Australia, in 2018, and the School of Information Technology, Deakin University, Australia, in 2019. He was a Research Fellow with the Department of Electronic Engineering, Tsinghua University, China, in 2020. He is currently a Postdoctoral Fellow with the National Institute of Informatics, Japan. His research interests include steganography, forensics, biometrics, cybersecurity, computer vision, computational linguistics, and artificial intelligence.

XU WANG received the Ph.D. degree from Feng Chia University, in 2022. He is currently a Lecturer with the School of Information Science and Engineering, University of Jinan, Jinan, China. His research interests include image processing, multimedia security, information hiding, and deep learning.
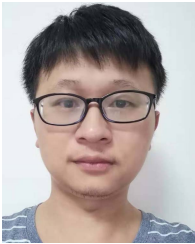
SISHENG CHEN received the B.S. degree from Shandong University, in 2005, and the M.S. degree in applied mathematics from Fujian Normal University, in 2008. He is currently pursuing the Ph.D. degree in information engineering and computer science with Feng Chia University, Taichung, Taiwan. He is also a Lecturer with Fujian Polytechnic Normal University. His research interests include multimedia security, image processing, and deep learning.

ISAO ECHIZEN received the B.S., M.S., and D.E. degrees from the Tokyo Institute of Technology, Japan, in 1995, 1997, and 2003, respectively. He joined Hitachi Ltd., in 1997, where he was a Research Engineer with the Systems Development Laboratory, until 2007. He was a Visiting Professor at Tsuda University, Japan, the University of Freiburg, Germany, and the University of Halle-Wittenberg, Germany. He is currently the Director and a Professor with the Information and Society Research Division, National Institute of Informatics (NII), the Director of the Global Research Center for Synthetic Media, NII, a Professor with the Department of Information and Communication Engineering, Graduate School of Information Science and Technology, The University of Tokyo, the Deputy Dean of the School of Multidisciplinary Sciences, Graduate University For Advanced Studies (SOKENDAI), Japan, and a Professor with the Department of Informatics, School of Multidisciplinary Sciences, SOKENDAI. He is also engaged in research on multimedia security and multimedia forensics. He also works as a Research Director of CREST FakeMedia Project, Japan Science and Technology Agency (JST). He was a member of the Information Forensics and Security Technical Committee of the IEEE Signal Processing Society. He received the Best Paper Award from the IPSJ, in 2005 and 2014, the Fujio Frontier Award and the Image Electronics Technology Award, in 2010, the One of the Best Papers Award from the Information Security and Privacy Conference, in 2011, the IPSJ Nagao Special Researcher Award, in 2011, the DOCOMO Mobile Science Award, in 2014, the Information Security Cultural Award, in 2016, and the IEEE Workshop on Information Forensics and Security Best Paper Award, in 2017. He is the Japanese Representative on IFIP TC11 (Security and Privacy Protection in Information Processing Systems), a Member-at-Large of Board-of-Governors of APSIPA, and an Editorial Board Member of the IEEE Transactions on Dependable and Secure Computing, the *EURASIP Journal on Image and Video Processing*, and the *Journal of Information Security and Applications* (Elsevier).

VICTOR SANCHEZ received the M.Sc. degree from the University of Alberta, Canada, in 2003, and the Ph.D. degree from The University of British Columbia, Canada, in 2010. From 2011 to 2012, he was with the Video and Image Processing Laboratory, University of California at Berkeley, as a Postdoctoral Researcher. In 2012, he was a Visiting Lecturer with the Group on Interactive Coding of Images, Universitat Autònoma de Barcelona. From 2018 to 2019, he was a Visiting Scholar with the School of Electrical and Information Engineering, The University of Sydney, Australia. He is currently an Associate Professor with the Department of Computer Science, University of Warwick, U.K. His research has been funded by the Consejo Nacional de Ciencia y Tecnología, Mexico; the Natural Sciences and Engineering Research Council of Canada; the Canadian Institutes of Health Research; the FP7 and the H2020 Programs of the European Union; the Engineering and Physical Sciences Research Council, U.K; and the Defence and Security Accelerator, U.K. His research interests include signal and information processing with applications to multimedia analysis, image and video coding, and security and communications. He has authored several technical articles and book chapters in these areas.

CHANG-TSUN LI received the B.Sc. degree in electrical engineering from the National Defence University, Taiwan, the M.Sc. degree in computer science from the U.S. Naval Postgraduate School, USA, and the Ph.D. degree in computer science from the University of Warwick, U.K. He is currently a Professor of cyber security with Deakin University, Australia. He has had over 20 years research experience in multimedia forensics and security, biometrics, machine learning, data analytics, computer vision, image processing, pattern recognition, bioinformatics, and content-based image retrieval. The outcomes of his research have been translated into award-winning commercial products protected by a series of international patents and have been used by a number of law enforcement agencies, national security institutions and companies around the world, including INTERPOL (Lyon, France), U.K. Home Office, Metropolitan Police Service (U.K.), Sussex Police Service (U.K.), Guildford Crown Court (U.K.), and U.S. Department of Homeland Security. The research team of the CSCRC-funded Development of Australian Cyber Criteria Assessment (DACCA) project under his leadership is the Winner of the 2021 AISA Cyber Security Researcher of the Year Award. He is a member of IEEE Information Forensics and Security Technical Committee. He is also the Chair of Computational Forensics Technical Committee of the International Association of Pattern Recognition (IAPR). He is an Associate Editor of IEEE Transactions on Circuit and System for Video Technology, *IET Biometrics*, and the *EURASIP Journal of Image and Video Processing*.

• • •