Cognitive Computation
https://doi.org/10.1007/s12559-024-10293-1

RESEARCH

Multi-resolution Twinned Residual Auto-Encoders (MR-TRAE)—A Novel DL Model for Image Multi-resolution

Alireza Momenzadeh[1,2] · Enzo Baccarelli[2] · Michele Scarpiniti[2] · Sima Sarv Ahrabi[3]

Received: 23 November 2023 / Accepted: 30 April 2024

Abstract
In this paper, we design and evaluate the performance of the Multi-resolution Twinned Residual Auto-Encoders (MR-TRAE) model, a deep learning (DL)-based architecture specifically designed for achieving multi-resolution super-resolved images from low-resolution (LR) inputs at various scaling factors. For this purpose, we expand on the recently introduced Twinned Residual Auto-Encoders (TRAE) paradigm for single-image super-resolution (SISR) to extend it to the multi-resolution (MR) domain. The main contributions of this work include (i) the architecture of the MR-TRAE model, which utilizes cascaded trainable up-sampling modules for progressively increasing the spatial resolution of low-resolution (LR) input images at multiple scaling factors; (ii) a novel loss function designed for the joint and semi-blind training of all MR-TRAE model components; and (iii) a comprehensive analysis of the MR-TRAE trade-off between model complexity and performance. Furthermore, we thoroughly explore the connections between the MR-TRAE architecture and broader cognitive paradigms, including knowledge distillation, the teacher-student learning model, and hierarchical cognition. Performance evaluations of the MR-TRAE benchmarked against state-of-the-art models (such as U-Net, generative adversarial network (GAN)-based, and single-resolution baselines) were conducted using publicly available datasets. These datasets consist of LR computer tomography (CT) scans from patients with COVID-19. Our tests, which explored multi-resolutions at scaling factors $\times(2, 4, 8)$, showed a significant finding: the MR-TRAE model can reduce training times by up to 60% compared to those of the baselines, without a noticeable impact on achieved performance.

Keywords Image multi-resolution · Semi-blind joint training · Training time vs. performance trade-off · Auxiliary multiple decoding output branches · Knowledge distillation · Teacher-student learning paradigm · Hierarchical cognition

Background, Motivations, and Goals

The aim of image super-resolution (SR) is to improve the visual quality of blurred and potentially noisy low-quality images. SR achieves this by generating one or more high-resolution (HR) versions of a given low-resolution (LR) image. The role of image SR is recognized across various domains, including biomedical research, surveillance, remote sensing, and medical diagnosis. In these fields, the enhancement from LR to HR images may play a key role, due to the constraints imposed by existing computational and communication resources [1].

In principle, image SR can be achieved through methods in either the hardware (HW) or software (SW) domains [2]. HW-based SR techniques offer rapid computation due to less reliance on software processing but often necessitate an increase in HW chip size or a reduction in pixel/sensor sizes. However, enlarging the chip size leads to higher costs and

✉ Alireza Momenzadeh
alireza.momenzadeh@uniroma1.it ;
alireza.momenzadeh@iit.cnr.it

Enzo Baccarelli
enzo.baccarelli@uniroma1.it

Michele Scarpiniti
michele.scarpiniti@uniroma1.it

Sima Sarv Ahrabi
sima.sarvahrabi@santannapisa.it

[1] Institute of Informatics and Telematics, National Research Council, Via Giuseppe Moruzzi 1, Pisa 56124, Italy

[2] Department of Information Engineering, Electronics and Telecommunications (DIET), Sapienza University of Rome, Via Eudossiana 18, Rome 00184, Italy

[3] Department of Excellence (EMbeDS), Sant'Anna School of Advanced Studies, Piazza Martiri della Libertà, 33, Pisa 56127, Italy

Published online: 21 May 2024

power consumption, while shrinking pixel or sensor sizes can decrease light intensity and increase shot noise. Moreover, HW-based SR approaches usually require costly and complex system redesigns [3]. As an alternative, SW-based SR techniques present a less expensive option that does not necessitate changes to the existing HW configuration [3].

Image SR techniques, whether based in HW or SW, can be categorized into multi-frame image super-resolution (MISR) or single image super-resolution (SISR) on the basis of the number of input images processed simultaneously [4]. MISR techniques enhance image resolution by combining various spatial views of the same scene. Conversely, SISR techniques generate one or more HR images, at multiple spatial resolutions, from a single LR input image. SISR is particularly relevant in application scenarios where multiple scene views are unavailable or when the temporal correlation between views is low [1]. However, increasing the spatial resolution of single images, such as de-blurring LR computer tomography (CT) scans, poses significant challenges due to the computing-intensive nature of the process. This complexity arises because the SR problem is inherently ill-posed; that is, it is an optimization problem that typically results in multiple solutions of differing visual quality [4]. Various traditional optimization techniques have been employed to address the SISR challenge, including non-linear regularization, filtering, wavelet-based, and statistical-based methods [5].

Finally, deep learning (DL)-based techniques inspired by biological processes have been gaining momentum in the SISR field. This is largely attributed to their ability to replicate human brain cognitive processes, which learn complex mappings from examples without the need for predefined formal models. This contrasts with traditional optimization-based approaches; DL-based methods try to emulate the reasoning processes of the brain. They autonomously learn the relationships between different datasets, enabling the extraction of spatial patterns and features that are difficult—or sometimes impossible—to model analytically [2, 6].

Classification of Current DL-Based Approaches for SISR

In principle, state-of-the-art DL-based SISR techniques are categorized into supervised, unsupervised, and domain-specific approaches [7]. Below, we provide a brief overview and comparative analysis of their respective pros vs. cons under the SISR realm.

In the *supervised* approach to SISR, DL models are trained using datasets containing *paired* LR and HR versions of each image. Through per-image comparison of these versions, the model learns to convert, during the testing phase, each LR image into one or more super-resolved outputs at various scaling factors. Although this method often results in images of high visual quality, its effectiveness depends on the availability of paired LR/HR images for training. Consequently, supervised SISR techniques are mainly applied in fields where high-quality HR images can serve as ground truth, such as in some medical applications [4].

The main feature of *unsupervised* SISR techniques, also known as *blind* SISR methods, is their ability to be trained *without* the need for paired LR/HR image datasets [4]. Depending on the implemented training approach, unsupervised SISR methods utilize (i) *unpaired* datasets of HR and LR images, so as to give rise to the so-called *weakly* unsupervised training, or (ii) datasets consisting of a *single* LR image, in order to extract intra-image statistics during the training [4]. Emerging unsupervised SISR methods increasingly incorporate the GANs [8]. While unsupervised SISR approaches do not require the utilization of training datasets, the visual quality of the rendered images does not match that achieved by their supervised counterparts [8].

Finally, *domain-specific* approaches to SISR are designed for specific types of images or application areas, such as satellite imagery SISR and facial imagery SISR. These methods stand out by using *specific* domain knowledge to fine-tune the loss functions used during training. By concentrating on particular domains, these models can deliver high performance within their intended application areas. However, they usually lack the ability to generalize across unrelated domains [7].

Each of the methodologies discussed—namely supervised, unsupervised, and domain-specific approaches—has its advantages and disadvantages in SISR. Supervised methods are known for their stable training processes but require paired LR and HR image datasets for training. Unsupervised SISR methods, while eliminating the need for paired datasets, may not achieve the same level of image quality as supervised techniques. That is particularly critical in medical applications where the accuracy of spatial details is more important. The training of unsupervised methods is more prone to instability and can result in super-resolved images damaged by artifacts in HR outputs. Domain-specific SISR models give great performance within their designated application areas, but not in applications outside their specialized domain.

Recent studies, such as those by [1, 7], highlight the potential of domain-oblivious semi-blind (i.e., hybrid supervised-unsupervised) training techniques for SISR applications. These methods aim to minimize the training dataset size without reducing the generalization performance of the SR models during testing. This paper takes such a design approach.

Motivations and Contributions

About the multi-resolution (MR) image paradigm, two questions arise: (i) its potential application fields and (ii) its advantages and disadvantages compared to other image-scaling techniques.

Application Fields Addressing the first question, the need for multiple spatial resolutions of a single ground-truth image forms the basis of various Information and Communication Technology (ICT) applications that depend on adaptable resources. The optimal selection of image resolution, as discussed in Chapters 4, 5, and 6 of [9], is important for (i) adapting image rendering to the computational capabilities of devices, which may vary over time or be initially unknown; (ii) adjusting variable-bit-rate (VBR) encoding to fit communication link capacities; (iii) adaptive image recording, to fit the storage capacities; and (iv) multi-spectral analysis of remote-sensing images, for multi-scale enhancement and feature extraction from low-resolution images [10]. More broadly, having images available at different scaling factors can [11] (i) reduce computational complexity by enabling multi-scale algorithmic processing, (ii) improve numerical robustness through multi-scaling transforms as algebraic preconditioners, (iii) simplify algorithms by revealing hidden features that may be easier to process, and (iv) make cognitive reasoning better by modeling or analyzing images across multiple spatial scales to uncover deeper insights into hidden features. As indicated in [11], such capabilities are critical in managing the complete life cycle (acquisition, processing, rendering, and storage) of medical images, including CT, X-ray, and magnetic resonance scans of considerable size.

Competing Approaches and Their Advantages and Disadvantages To the best of the authors' knowledge, mainly two strategies exist for generating multiple scaled versions of an image: the multiple-single resolution (M-SR) approach and the multi-resolution (MR) approach [11]. M-SR sequentially applies a single-resolution network multiple times to the same input for different scales. Conversely, MR uses a singular network to simultaneously produce in parallel all scaled versions in a one-shot way. This design choice enables MR to utilize computational resources and share parameters across different scales. As mentioned by [12], two advantages favor the MR approach over M-SR. First, our analysis in "Complexity Analysis and Implementation Aspects" section demonstrates that DL-based MR models are generally less complex and quicker to train than training single-resolution models multiple times. Secondly, the perceived quality of images by humans does not always align with numerical performance metrics, such as classification accuracy or peak signal-to-noise ratio, assessed by automated systems. This requires that multiple resolutions of a single LR image must be compared by considering different performance metrics [12].

Domain-Oriented Classification of General Image MR Methods MR methods can be categorized into two main types based on the domain and multi-resolution strategy used [11]:

(i) Wavelet-based methods employ wavelet transforms in spatial or frequency domains to break down an image into various scales, mainly for image de-noising or segmentation.
(ii) Hierarchical methods progressively divide a ground-truth image into components of different resolutions. These methods use the concept of hierarchical cognition, creating simpler models at various scales that may be combined to make the comprehension of a complex model easier.

Inspired by the aforementioned considerations, the main goal of this paper is to design and evaluate the MR-TRAE model, a neural network for multi-resolution image processing influenced by the hierarchical MR paradigm. The MR-TRAE model uses semi-blind training to SISR and extends the TRAE concept, previously introduced by the authors of [13] for single-resolution SISR, to multi-resolution applications.

Therefore, motivated by these considerations, the main goal of this paper is to design and test the performance of the MR-TRAE model. This is a neural networking architecture "ad hoc" designed for image *multi-resolution* and inspired by the (above mentioned) *hierarchical* MR paradigm. Specifically, the MR-TRAE model relies on *semi-blind* training for attaining SISR and *generalizes* to the multi-resolution realm of the TRAE paradigm recently proposed by the authors in [13] for the more specific case of single-resolution SISR.

This paper presents several contributions towards the development of the MR-TRAE model:

(i) We have developed the MR-TRAE model to extend the TRAE model [13] by (i) integrating cascading upsampling modules for scaling output images and (ii) using auxiliary Auto-Encoder (AE) output branches—which act as implicit teachers—for model training.
(ii) A novel loss function has been introduced for simultaneously training all components of the MR-TRAE. This semi-blind training approach does not require ground-truth images at the intermediate resolutions, making it unique compared to our earlier single-resolution model [13]. Training with only the lowest and highest resolution image pairs is a distinctive aspect of the MR-TRAE model.
(iii) The performance of MR-TRAE is evaluated against leading models such as U-Net [14], multiple single-resolution TRAE (M-SR-TRAE) [13], and super-resolution GAN (SRGAN) [15], using open-access datasets of variable-size CT scans for COVID-19 [16]. This ensures a fair comparison of its effectiveness.
(iv) Our findings show that the MR-TRAE model reduces training times by up to 60% relative to baseline models without affecting test performance. This demonstrates an effective balance between model simplicity and efficacy.

(v) Additionally, we highlight the MR-TRAE model's contributions to cognitive-inspired areas such as knowledge distillation, the teacher-student learning paradigm, and hierarchical cognition that present its broader relevance to these fields.

The structure of this paper is as follows: "Related Work" section reviews relevant literature related to our contributions. "Problem Statement and Pursued Solving Method" section details the MR-TRAE model, including its architecture, specially designed training functions, semi-blind training approach, and a complexity analysis alongside implementation insights. "MR-TRAE Novelties and Related Cognitive Aspects" section explores the innovations of MR-TRAE and its relations to cognitive paradigms. "Experimental Setup" section outlines the experimental framework, including the simulation of DL models and the datasets used for training and testing. "Performance Results and Comparisons" section presents a performance evaluation of MR-TRAE and compares it with established baselines. "Conclusion and Hints for Future Research" section summarizes the key findings and suggests directions for future research.

Related Work

The field of SISR embraces diverse (often heterogeneous) methodologies with extensive research from various aspects. Our MR-TRAE model mainly utilizes convolutional neural networks (CNNs) as its core components. Therefore, this review concentrates on the latest CNN-based models developed for SISR. For a more comprehensive exploration of SISR, readers may consult recent surveys such as those in [2, 4, 7].

The literature on SISR can be broadly categorized into two interconnected research areas, as discussed in [4]: CNN-based model architectures for SISR and domain-specific applications of CNN-based SISR architectures.

CNN-based architectures for SISR Applications of SISR often depend on complex DL models designed for intensive image processing tasks. A central question that logically connects much of the research on DL architectures for SISR is how to minimize model complexity while maintaining high quality of rendered image.

Early research in the field of SISR is directed towards assessing the effectiveness of basic CNN models. The work in [17] introduces the single-resolution CNN (SR-CNN) for SISR and sets a foundational benchmark despite its moderate image quality improvements [7]. Building on the SR-CNN, the authors of [18] develop the multi-scale SR (MDSR) network and improve the original architecture by simplifying some non-linear components. This adjustment aims to stabilize the training process and to improve visual quality across various spatial resolutions. A limitation of these models is their relatively shallow architecture that limits their effective multi-resolution scaling capabilities to a maximum of $\times 4$.

Caused ed by these limitations, a second line of research emerged, focusing on the development of residual and, potentially, dense CNN-based models [19] for SISR. The aim is to make the network architecture deeper, while preventing training instability. The success of DenseNet [20] in achieving high classification accuracy inspired the creation of several SR algorithms using densely connected CNNs to enhance feature extraction [19, 20]. Liu et al. [21] explored this by using the hierarchical structure of residual branches to incorporate multiple convolutional layers with strategic skip connections to ease better information flow. However, it has been observed that the performance of such architectures becomes worse rapidly with scaling factors above $\times 4$.

In response to these problems, [22] introduced the MASA network for integrating a module that enables coarse-to-fine spatial feature mapping and a spatial adaptive module for aligning feature distribution with that of LR input images. Similarly, [23] developed the DeFiAN model, a CNN-based architecture that employs a Hessian filter to identify high-frequency features for refinement through an SR encoder-decoder process. Then, [24] introduced a degradation-aware SR (DASR) capable of identifying various degradation patterns to improve SR performance by learning distinct feature representations. Despite the fact that these models gained state-of-the-art results, they still struggle to find an optimal balance between model complexity and test performance, particularly at higher scaling factors such as $\times 8$ [4].

Relating to balance between model complexity and performance, a third research direction focused on the application of attention mechanisms to enhance the efficacy of baseline Res/DenseNet architectures [2]. Notable examples include RCAN [25], SAN [26], HAN [27], and RFANet [21]. The design of these models was based on using attention mechanisms to make the network deeper and to increase image features through strategic cross-channel and cross-layer interactions. In line with this, [28] proposes a series of scalable architectures that integrate densely connected networks with attention mechanisms to reduce overfitting. At the same time, [29] developed an adaptive attention module aimed at improving the reconstruction of high-frequency details. Additionally, [30] introduces IDSRN, which achieves multi-scale feature extraction via carefully designed attention mechanisms. Despite the top-tier performance of these attention-enhanced models, the balance between model complexity and performance efficiency remains less than ideal [4].

Recent advancements in the literature [31] have led to the development of the multi-scale fractal residual attention networks (MS-FR-ANs). This approach joins fractal residual blocks and advanced channel attention mechanisms together to enable adaptive multi-scale feature extraction and

improve the efficiency of inter-layer information transfer. An important implementation of the MS-FR-AN concept is the multi-scale information distillation network (MIDN) detailed in [32]. MIDN combines a fractal multi-scale feature distillation block with a variable-size kernel attention block, aiming for superior performance in handling multi-resolution images. But, the incorporation of fractal blocks makes challenges in achieving stable training, as stated in [33].

SISR Architectures for Domain-Specific Applications In the domain of SR for *natural* images, the architecture presented in [34] is designed to increase inference efficiency by identifying primary and secondary spatial features via trainable spatial masks to optimize the equilibrium in model complexity-performance. The CFSRCNN framework [35, 36] and the ESRT model [37] use cascaded CNN-based structures to ease image upscaling through transformer mechanisms. Then, [38] employs a learning-based approach for 3D EPI restoration, and [39] focuses on edge detection by integrating multi-resolution feature extraction and fusion. Additionally, [40] uses auxiliary semantic segmentation networks to guide SR learning processes and to improve texture details and color accuracy.

In the field of SR for *biomedical* images, [10] introduces a dual-branch network that effectively combines residual blocks via information distillation to improve image quality. Kong et al. [41] presents a supervised multi-stage training strategy and incorporates a loss function to enhance the visual quality of super-resolved medical images. The work in [42] proposes a non-linear perceptual multi-scale network to optimize the model complexity-performance balance. This network shows a multi-cascade residual nested-group module designed for extracting diverse image features across multiple spatial scales. This configuration enables dynamic selection and fusion of spatial features and improves the visual quality of the reconstructed images.

A novel line of research such as [8], [43], and [44] have focused on using DL models to classify CT and X-ray scans of patients with COVID-19. The aim is to identify and extract hidden features within the images that may not be readily observable or identifiable by medical professionals. This approach is formed based on the concept that multi-layered DL models simulate the human brain's hierarchical and layered processing of input data and give insights that are not immediately apparent through traditional medical analysis.

The study in [43] offers a review of DL-based approaches for detecting COVID-19 using chest X-rays and CT scans. It includes a detailed performance comparison of four DL models: VGG16, VGG19, ResNet50, and DenseNet by using publicly available COVID-19 CT and chest X-ray datasets. Based on their experimental findings, [43] states that the VGG19 model performs better than the others in detection accuracy. Sarv Ahrabi et al. [8] and Goel et al. [44] explore a different way by focusing on the application of GANs for classifying COVID-19 diseases from CT scans and chest X-ray images, marking a shift towards more innovative use of DL models.

Relating to the issues of limited datasets, [44] introduces a GAN-based architecture capable of generating HR synthetic CT images. To increase the performance of the GAN generator, the whale optimization algorithm (WOA) is used for hyper-parameter optimization. The performance metrics of the optimized GAN model indicate its superiority over various state-of-the-art meta-heuristic approaches, including genetic algorithms, pattern search, particle swarm optimization, simulated annealing, and Grey-Wolf optimization. Furthermore, [8] aims to evaluate the effectiveness of hidden features produced by the encoders of two advanced GAN architectures: Bidirectional GANs (BiGANs) and CycleGANs, in classifying COVID-19 diseases from CT scans. The findings show that, while CycleGAN-based models have the highest classification accuracy among the tested frameworks, they increase training duration and model complexity.

Overall, Table 1 offers a concise overview of the research discussed to summarize the key aspects, methodologies, and findings of the studies reviewed.

MR-TRAE Positioning in the Current SISR Research Realm Based on our review, the MR-TRAE model, as depicted in Fig. 2, introduces three novelties. First, it takes the innovative concept of "twinned" auto-encoders (AEs) from [13] to achieve MR image processing. Second, based on knowledge distillation and hierarchical cognition principles, MR-TRAE employs hierarchically structured output branches from the intermediate layers of the AEs as "teacher" (reference) signals, as sketched in the lower part of Fig. 2. This approach allows the semi-blind training of the overall MR-TRAE network. Lastly, the designed MR-TRAE's training methodology allows to bypass the need for multiple datasets or training stages at different resolution scales. Instead, a single training session using just two paired HR and LR datasets is sufficient, as remarked by the HR-LR "paired" input configuration in Fig. 2. This makes the training process less cumbersome, while maintaining effective learning and adaptation across various resolution scales. Overall, to the best of the authors' knowledge, the use of the outputs of the intermediate hidden layers of an AE's decoder as reference signals is the main architectural novelty of the proposed MR-TRAE model.

This review shows that the MR-TRAE model is unique in terms of possessing the aforementioned architectural features.

Table 1 A synoptic view of the reviewed CNN-based SISR research

Work	Acronym	Pursued approach to SISR
[17]	SR-CNN	Three-layer CNN with pre up-sampling
[18]	MDSR	Optimized multi-scale deep residual CNN without using batch normalization
[19]	RDN	Residual CNN equipped with dense skip connections
[29]	AMSRN	Attention-based multi-resolution residual network
[26]	SAN	Second-order attention-based residual CNN
[21]	RFA	Residual feature aggregation model
[30]	IDSRN	Dual-scale residual CNN network
[45]	WMRN	Weighted multi-resolution residual CNN-based network
[36]	ACNet	Asymmetric deep CNN
[38]	3DVSR	Dual stage image up-sampling
[39]	Cross-SRN	Cross CNN blocks plus multi-resolution feature extractor
[40]	SSG-RWSR	Residual dense blocks plus segmentation network
[41]	RLFN	Residual deep CNN based on feature distillation blocks
[31]	MFRAN	Multi-scale fractal residual attention-based network
[46]	MEM	Deep CNN equipped with multi-resolution enhancement modules
[32]	MSID	Deep CNN based on multi-resolution receptive field and variable-size kernel attention
This work	MR-TRAE	Joint twinned AEs equipped with multi-scale CNN-based up-samplers

Problem Statement and Pursued Solving Method

This section aims to achieve four objectives. Initially, we introduce the MR-TRAE model and clarify how it is different from the single-resolution version introduced in [13]. Next, we elaborate on the loss functions and outline the primary steps of the semi-blind training methodology devised for the MR-TRAE model. Then, we explore the complexity of the MR-TRAE model and cover key aspects of its implementation. Lastly, we identify various potential applications for the MR-TRAE model during the test phase.

At this point, we briefly revisit the fundamental elements of the previously developed single-resolution TRAE architecture to be able to describe the improvements in the MR-TRAE model.

An Overview of the Foundational TRAE Architecture

The SISR paradigm establishes a mapping between pairs of LR and HR images of the same scene. A conventional AE has only one input image at a time to learn a compressed yet informative representation of it (known as hidden features) and to recreate a nearly perfect output from this compressed form. Thus, a standard AE is not designed to map between pairs of LR/HR images. To overcome this limitation and apply the AE concept to SISR, [13] introduced an advanced version of the AE architecture, termed the "twinned" AE. Figure 1 presents a simplified diagram of the TRAE architecture developed in [13] for single-resolution image processing at the training phase (refer to [13] for a comprehensive discussion on the TRAE architecture during both training and testing phases).

The characteristic of the basic single-resolution TRAE is its "twinned" structure. This setup shows a "Follower AE" and a "Master AE" that operate together on the LR and HR versions of the same input image, respectively. Each AE in this architecture is tasked with encoding its respective input image into a compressed format to reconstruct the original image from this compact representation.

In [13], the authors highlight that the unique feature of the TRAE model is its training methodology, which develops a cooperative interaction between the Master and Follower AEs. This collaboration enables the Master AE to share insights learned from its HR input with the Follower AE. The aim is to improve the quality of the SR image produced by the Follower AE using the information received from the Master AE. Through this process, both AEs engage in a form of transfer learning, exchanging important information to gradually minimize the differences in their generated hidden features. This collaborative training strategy, as supported by theoretical analysis and empirical evidence in [13], improves the TRAE's ability to accurately map between LR and HR images.

The foundational components of the TRAE model depicted in Fig. 1 include the following:

(i) *Master AE*: This serves dual purposes. It extracts hidden features from the HR input image and, in addition, aims to output a version very similar to its input. The Master AE functions like a classical AE, trained on a dataset of HR images.

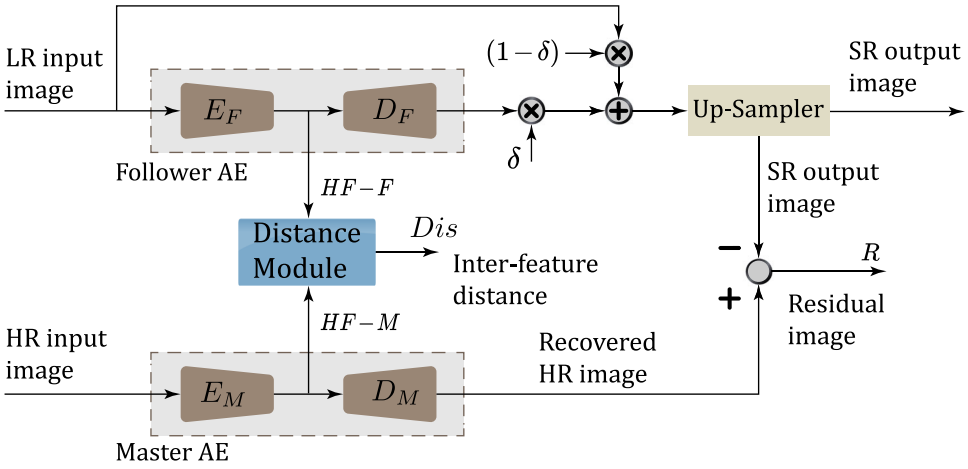


Fig. 1 A (simplified) sketch of the TRAE model architecture in [13], referring to the training phase. During the test, only the trained Follower AE, skip connection, and up-sampler module are retained. HR, high resolution; LR, low resolution; F, Follower; M, Master; Dis, distance; R, residual; E/D, encoder/decoder; HF-F, hidden features from the Follower AE; HF-M, hidden features from the Master AE

(ii) *Follower AE*: This has two functions. It extracts hidden features from the LR input image and produces a residual output version of its LR input.

(iii) *Up-sampling Module*: This is located at the output of the Follower AE. Its role is to up-sample the LR image output from the summation node depicted in Fig. 1 and to produce the super-resolved version of the corresponding LR input image.

(iv) *Residual module*: This calculates the element-wise difference between the images of the same size produced by the up-sampling module and the Master AE. The resultant residual image serves as the objective to minimize throughout the training process of the entire TRAE model.

(v) *Distance module*: This calculates the distance between the feature vectors of identical size produced by the Follower and Master AEs. The computed inter-feature distance, denoted as Dis, serves as a metric for guiding the training of the TRAE model. The choice of distance metric can vary based on the application domain. For an exploration of how the TRAE sensitivity is influenced by the selected inter-feature distance metric, refer to [13].

(vi) *Global skip connection*: This features an adjustable gain parameter, δ, and is designed to enable the Follower AE to learn a residual version of the LR input image rather than the full LR input itself. The gain parameter δ, which ranges from $0 \leq \delta \leq 1$, is optimized during training to achieve an optimal balance between the relative contributions of the Follower AE's feedforward and residual branches.

The Proposed MR-TRAE Model: Architecture, Loss Functions, and Training Procedure

Architecturally, the MR-TRAE model improves the TRAE framework by incorporating auxiliary output branches within the Master AE. This modification allows for the simultaneous production of multiple super-resolved images in a single inference operation.

To describe the MR-TRAE, we begin by defining a spatial image with M rows, N columns, and C color channels as a tensor I of dimensions $M \times N \times C$, meaning $I \in \mathbb{R}^{M \times N \times C}$. For the sake of simplicity, we will use a vector representation of an image I in subsequent discussions. Therefore, $\vec{I}$ denotes a real-valued, column-wise vector of the image, having a dimension of $(MNC \times 1)$.

In conventional SISR methods, a network takes an LR input image, denoted as $\vec{I}_{LR} \in \mathbb{R}^{M_L N_L C \times 1}$, and aims to produce an SR output image, $\vec{I}_{SR} \in \mathbb{R}^{M_S N_S C \times 1}$, an SR version of a ground-truth HR image—represented by $\vec{I}_{HR} \in \mathbb{R}^{M_H N_H C \times 1}$—where the dimensions satisfy $M_L \leq M_S \leq M_H$ and $N_L \leq N_S \leq N_H$.

Typically, the LR and HR versions of the same image are linked by a transformation that is often "a priori" unknown or stochastic, as discussed in [1]:

$$\vec{I}_{LR} = F(\vec{I}_{HR}), \tag{1}$$

where $F(\cdot)$ represents the HR-to-LR mapping function.

Figure 2 shows the architecture of the MR-TRAE during its training phase. Next, we will explain the collections of LR and HR images used for training, along with the functions and significance of the various components within the MR-TRAE architecture.

Paired HR and LR Training Sets The set $S_{HR} = \left\{\vec{I}_{HR,n} \in \mathbb{R}^{M_H N_H C \times 1} \quad n = 1, 2, \cdots, T\right\}$ of size T gathers the $(M_H N_H C)$-dimensional HR image vectors used for the training. The paired set of LR training image vectors is denoted by $S_{LR} = \left\{\vec{I}_{LR,n} \in \mathbb{R}^{M_L N_L C \times 1} \quad n = 1, 2, \cdots, T\right\}$. As depicted in Fig. 2, paired HR and LR images act as parallel inputs to the Master and Follower AEs, respectively.

Master AE The Master AE consists of encoding (E_M) and decoding (D_M) deep neural networks, with their trainable

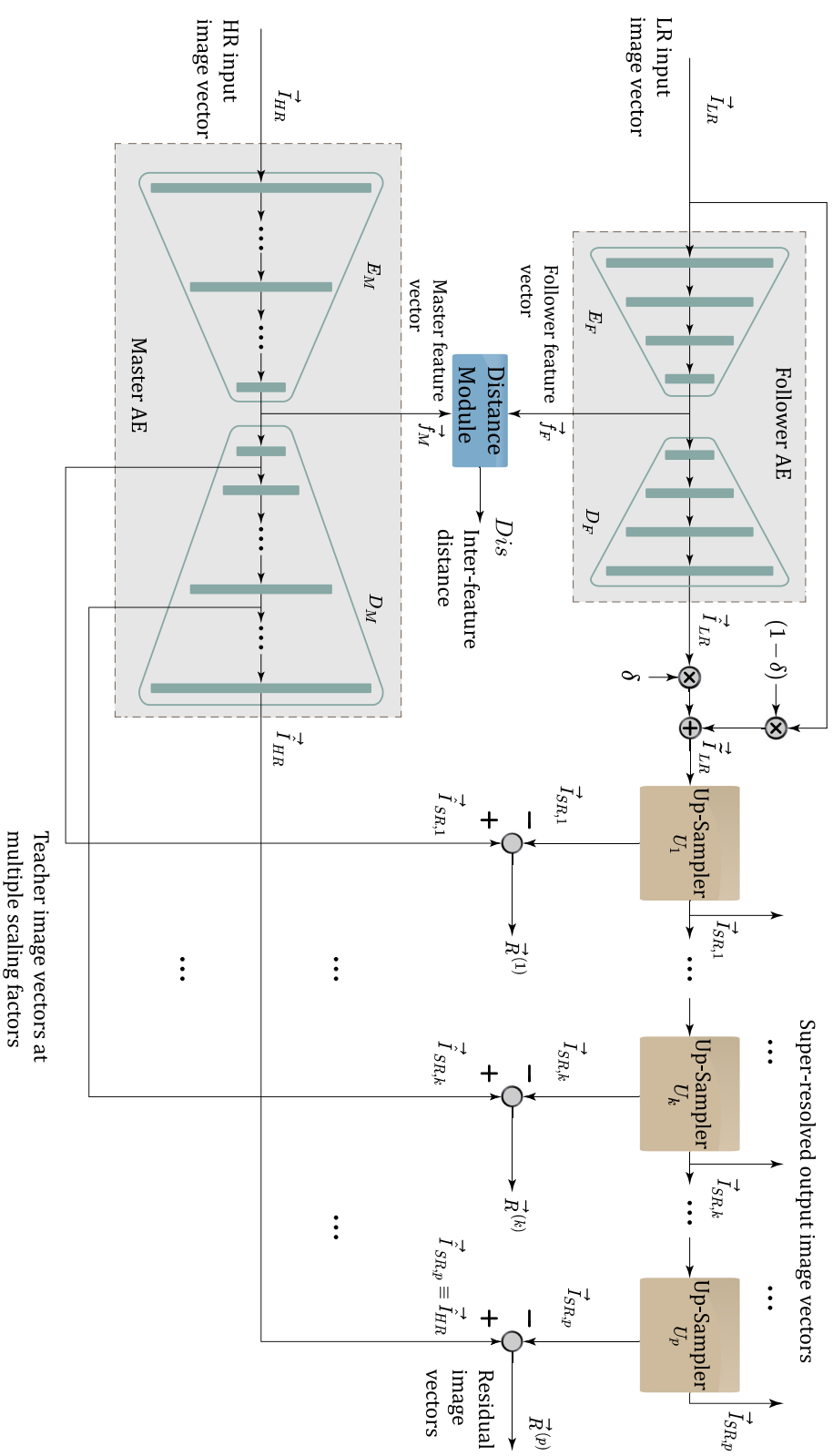


Fig. 2 The MR-TRAE architecture in the training phase. In the test phase, the skip connection with the trained gain δ, the trained Follower AE, and one or more trained up-samplers are retained, so as to generate in parallel one or more output images at the desired multi-resolution scaling factors in correspondence of each LR test image given in input to the Follower AE. E_M/D_M, Master Encoder/Decoder; E_F/D_F, Follower Encoder/Decoder; f, feature; R, residual; Dis, inter-feature distance

parameters represented by the vectors $\vec{\theta}_{E_M}$ and $\vec{\theta}_{D_M}$, respectively. Therefore, the HR output vector $\vec{\hat{I}}_{HR,n}$, produced by the Master AE in response to the HR training input vector $\vec{I}_{HR,n}$, is formulated as follows:

$$\vec{\hat{I}}_{HR,n} = D_M\Big(E_M\Big(\vec{I}_{HR,n}; \vec{\theta}_{E_M}\Big); \vec{\theta}_{D_M}\Big). \tag{2}$$

A distinction between the TRAE and MR-TRAE models is that in the MR-TRAE framework, the hidden layers of the Master Decoder are designed to offer auxiliary parallel outputs. As shown in Fig. 2, the vector $\vec{\hat{I}}^{(k)}_{SR,n}$ produced by the k-th hidden layer of the Master Decoder is expressed as follows:

$$\vec{\hat{I}}^{(k)}_{SR,n} = D_M^{(1:k)}\Big(E_M\Big(\vec{I}_{HR,n}; \vec{\theta}_{E_M}\Big); \vec{\theta}_{D_M}\Big), \quad k \geq 1, \tag{3}$$

where $D_M^{(1:k)}(.;.)$ denotes the transformation induced by the cascade of the first k hidden layers of the Master Decoder. In our configuration, the hidden layers of the decoder are sequentially numbered from left to right in Fig. 2, with the innermost layer of the decoder designated as $k = 1$.

In the MR-TRAE model, the auxiliary output vector $\vec{\hat{I}}^{(k)}_{SR,n}$ is designed to have a size that lies between those of the HR and LR input vectors. Its size progressively increases as one moves from the innermost to the outermost layer of the Master Decoder. Consequently, the spatial resolution of the image vector $\vec{\hat{I}}^{(k)}_{SR,n}$, produced by the k-th output branch of the Master Decoder, exceeds that of the image vector $\vec{\hat{I}}^{(k-1)}_{SR,n}$ generated by the preceding $(k-1)$-th output branch, as depicted in Fig. 2.

Follower AE and Skip Connection The Follower AE consists of the encoding (E_F) and decoding (D_F) neural networks, with trainable parameters gathered in $\vec{\theta}_{E_F}$ and $\vec{\theta}_{D_F}$, respectively. The output of the Follower AE is the residual LR image vector $\vec{\hat{I}}_{LR,n}$, which is connected to the corresponding LR input image vector $\vec{I}_{LR,n}$ as follows:

$$\vec{\hat{I}}_{LR,n} = D_F\Big(E_F\Big(\vec{I}_{LR,n}; \vec{\theta}_{E_F}\Big); \vec{\theta}_{D_F}\Big). \tag{4}$$

Therefore, the reconstructed LR image vector $\vec{\tilde{I}}_{LR,n}$, found at the output of the skip connection, is defined as follows:

$$\vec{\tilde{I}}_{LR,n} = (1-\delta)\,\vec{I}_{LR,n} + \delta\,\vec{\hat{I}}_{LR,n}. \tag{5}$$

Distance Module The function of the distance module $d(\cdot,\cdot)$ is to compute the scalar distance Dis_n between the matching-sized hidden feature vectors $\vec{f}_{M,n}$ and $\vec{f}_{F,n}$, produced respectively by the Master and Follower AEs in response to the paired LR/HR input vectors $\vec{I}_{LR,n}$ and $\vec{I}_{HR,n}$ as follows:

$$Dis_n = d\Big(\vec{f}_{M,n}, \vec{f}_{F,n}\Big), \tag{6}$$

with

$$\vec{f}_{M,n} = E_M\Big(\vec{I}_{HR,n}; \vec{\theta}_{E_M}\Big),$$

and

$$\vec{f}_{F,n} = E_F\Big(\vec{I}_{LR,n}; \vec{\theta}_{E_F}\Big). \tag{7}$$

Up-sampling and Differential Modules A series of $p \geq 2$ up-sampling modules is positioned at the output of the Follower AE, with the task of incrementally increasing the size of the LR image vector $\vec{\tilde{I}}_{LR,n}$ that emerges from the skip connection. Their purpose is to produce, at the same time, a set $\{\vec{I}^{(k)}_{SR,n}, k = 1, ..., p\}$ of SR output image vectors. The SR image vector $\vec{I}^{(k)}_{SR,n}$, created by the k-th up-sampling module, is defined as follows:

$$\begin{aligned} &\vec{I}^{(k)}_{SR,n} = [U_k]\,\vec{I}^{(k-1)}_{SR,n}, \quad k = 1, 2, \cdots, p, \\ &\vec{I}^{(0)}_{SR,n} \equiv \vec{\tilde{I}}_{LR,n}, \end{aligned} \tag{8}$$

where $[U_k]$ denotes the up-sampling matrix for the k-th module. The role of the k-th differential module is to compute the residual image vector $\vec{R}^{(k)}_n$ by calculating the difference between the vectors of the same size that are produced by the k-th output branch of the Master Decoder and the corresponding k-th up-sampling module, as illustrated in (3) and (8)

$$\vec{R}^{(k)}_n = \vec{\hat{I}}^{(k)}_{SR,n} - \vec{I}^{(k)}_{SR,n}, \quad k = 1, 2, \cdots, p. \tag{9}$$

We note that the aforementioned residual vectors facilitate the *semi-blind* training of the MR-TRAE.

Semi-blind MR-TRAE Training—Loss Functions and Training Procedure

The training sets S_{HR} and S_{LR} are paired; therefore, each LR image input $\vec{I}_{LR,n}$ at the Follower AE corresponds to the HR version $\vec{I}_{HR,n}$ at the Master AE. The logic—that supports the devised training loss functions and the associated semi-blind training procedure—is based on the following considerations.

On training convergence, the Master AE's output $\vec{\hat{I}}_{HR,n}$ should closely match its corresponding input $\vec{I}_{HR,n}$. This outcome lays the groundwork for the expectation that the auxiliary outputs of the Master Decoder will produce good quality SR versions $\{\vec{\hat{I}}^{(k)}_{SR,n}, k = 1, ..p\}$ of the input HR image $\vec{I}_{HR,n}$, with spatial dimensions intermediate to those of the HR and LR images used in training. This suggests the possibility of using the Master Decoder's auxiliary outputs as

reference signals for a training process similar to supervised learning for the corresponding Follower AE and the series of up-samplers.

Master AE Training Loss Function Given these considerations, the goal of the Master AE is to output a recovered HR image $\vec{\hat{I}}_{HR,n}$ that closely resembles the corresponding input $\vec{I}_{HR,n}$. Therefore, the loss function $\mathcal{L}_M$ for training the Master AE should rely only on the HR input image vectors and the trainable parameters of the Master Encoder/Decoder. Thus, it can be formulated as the distance between the input $\vec{I}_{HR}$ and the recovered $\vec{\hat{I}}_{HR}$ image vectors produced by the Master AE, i.e.,

$$\mathcal{L}_M \equiv \mathcal{L}_M\left(\vec{\theta}_{E_M}, \vec{\theta}_{D_M}\right) = \frac{1}{M_H N_H C}\left\|\vec{I}_{HR} - \vec{\hat{I}}_{HR}\right\|, \quad (10)$$

We emphasize that the formulation in (10) enables the Master AE to undergo training independently, without any reliance on the Follower AE, thereby justifying the designation "Master."

Follower AE Training Loss Function The concept driving the design of the loss function for the Follower AE's training is based on three considerations.

First, given that the Master AE's inputs are HR and of high quality, the information decoded by the Master Decoder potentially improve the quality of what the Follower AE can *independently* derive from its LR inputs, which are of lower quality.

Second, the Master AE can pass two kinds of information: (i) *perceptual information*, through hidden feature vectors $\vec{f}_{M,n}$, and (ii) *content information*, via intermediate SR output image vectors $\{\vec{\hat{I}}^{(k)}_{SR,n}, k = 1, ..., p\}$, as outlined in [15] for the terminology used.

Third, the loss function in (10) exclusively considers the trainable parameters of the Master AE; therefore, the loss function for the Follower AE's training should accordingly incorporate the Follower AE's trainable parameters $\vec{\theta}_{E_F}$ and $\vec{\theta}_{D_F}$, the adjustable gain δ of the skip connection, and the set of trainable parameters $\{[U_k], k = 1, ..., p\}$ of the up-sampling modules.

Based on these insights, we intend to construct the training loss function for the Follower AE, $\mathcal{L}_F$, as a convex combination of a perceptual training loss $\mathcal{L}_P$ and a content training loss $\mathcal{L}_C$. This can be expressed as follows:

$$\mathcal{L}_F = \gamma\mathcal{L}_C + (1-\gamma)\mathcal{L}_P, \quad (11)$$

where $\gamma \in (0, 1)$ represents a trade-off hyper-parameter, the optimal value of which is determined through experimental trials. Further details on this parameter and its tuning will be discussed in "Complexity Analysis and Implementation Aspects" section.

Since the goal of the perceptual loss, $\mathcal{L}_P$, is to align the hidden features produced by the Follower AE with those extracted by the Master AE, we define $\mathcal{L}_P$ as follows:

$$\begin{aligned}\mathcal{L}_P &\equiv \mathcal{L}_P\left(\vec{\theta}_{E_F}\right) = d\left(\vec{f}_M, \vec{f}_F\right)\\ &\equiv d\left(\vec{f}_M, E_F\left(\vec{I}_{LR}, \vec{\theta}_{E_F}\right)\right).\end{aligned} \quad (12)$$

The role of the distance module depicted in Fig. 2 is to implement the inter-feature distance function d, as specified in (12).

The content loss, $\mathcal{L}_C$, functions within the spatial domain and aims to minimize the overall difference between the SR images $\{\vec{I}^{(k)}_{SR}\}$ produced by the up-samplers and the respective images $\{\vec{\hat{I}}^{(k)}_{SR}\}$ generated by the auxiliary output branches of the Master Decoder:

$$\begin{aligned}\mathcal{L}_C &\equiv \mathcal{L}_C\left(\vec{\theta}_{E_F}, \vec{\theta}_{D_F}, \delta, \{[U_k], k = 1, ..., p\}\right)\\ &= \sum_{k=1}^{p} \frac{1}{M_k N_k C}\left\|\vec{R}^{(k)}\right\|,\end{aligned} \quad (13)$$

where $\vec{R}^{(k)}$ represents the k-th residual image vector as defined in (9), and $M_k N_k C$ denotes its column size.

Ultimately, the equations presented in (11), (12), and (13) show that the training phase is designed to ensure the Follower AE closely "mirrors" the Master AE's behavior, and this validates the use of the term "Follower."

Overall Loss Function and Related Semi-blind Training Procedure By definition, the total training loss function $\mathcal{L}_{MR-TRAE}$ for MR-TRAE is the sum of the training loss functions for the Master and Follower AEs. Therefore, it can be expressed as follows, by referring to (10) and (11):

$$\mathcal{L}_{MR-TRAE}\left(\Omega_{TR}\right) = \mathcal{L}_M + \mathcal{L}_F, \quad (14)$$

where

$$\Omega_{TR} = \{\vec{\theta}_{E_M}, \vec{\theta}_{D_M}, \vec{\theta}_{E_F}, \vec{\theta}_{D_F}, \delta, \{[U_k], k = 1, ..., p\}\}, \quad (15)$$

is the overall set of trainable MR-TRAE parameters.

For the training of the model, we have developed an iterative *semi-blind* procedure, based on two considerations. Firstly, the training of the Master and Follower loss functions, $\mathcal{L}_M$ and $\mathcal{L}_F$ as specified in (10) and (11) respectively, requires only the first two elements for the Master AE, and all elements for the Follower AE, from the set in Eq. (15). Secondly, training the Master loss function exclusively utilizes the HR training set S_{HR}. In contrast, the training of the Follower AE uses not only the LR training set S_{LR} but also

incorporates the sets of perceptual $\{\vec{f}_M\}$ and content $\{\vec{\hat{I}}_{SR}^{(k)}\}$ information vectors produced by the Master AE.

Based on these considerations, for each paired HR and LR training input image vectors $\{\vec{I}_{HR}, \vec{I}_{LR}\}$, the established training procedure follows these sequential steps:

Step 1: Beginning with the current parameter values, the Master AE updates its Encoder/Decoder parameters $\{\vec{\theta}_{E_M}^*, \vec{\theta}_{D_M}^*\}$ by training its loss function as described in (10) over several iterations. This optimization can be performed using any optimizer, such as stochastic gradient descent (SGD) or Adam.

Step 2: Utilizing the newly updated parameters $\{\vec{\theta}_{E_M}^*, \vec{\theta}_{D_M}^*\}$ and the current HR input training image vector $\vec{I}_{HR}$, the Master AE refreshes its hidden feature vector $\vec{f}_M^*$ and the set of intermediate super-resolved output image vectors $\{\vec{\hat{I}}_{SR}^{(k)*}, k = 1, \ldots, p\}$. These updated vectors are then passed to the Follower AE for further processing.

Step 3: The Follower AE refines its loss function as detailed in (11) through several iterations. The aim is to update its Encoder/Decoder parameters, the gain of the skip connection, and the up-sampling matrices: $\{\vec{\theta}_{E_F}^*, \vec{\theta}_{D_F}^*, \delta^*, \{[U_k^*], k = 1, ..., p\}\}$. For this task, the Follower AE uses the current LR input image vector $\vec{I}_{LR}$ in conjunction with the most recently updated perceptual $\vec{f}_M^*$, and content $\{\vec{\hat{I}}_{SR}^{(k)*}\}$ information vectors provided by the Master AE. These vectors serve as constant reference signals throughout the Follower AE's iterative training process.

These steps are cyclically repeated for each pair of HR and LR training input images until the convergence of the total MR-TRAE training loss function, as specified in (14), is achieved.

Before moving forward, it is important to provide two clarifications relating to the MR-TRAE training procedure.

First, the *semi-blind* characteristic of the training procedure is established in step 2. This step carries out the on-the-fly creation of auxiliary SR image vectors $\{\vec{\hat{I}}_{SR}^{(k)*}, k = 1, ..., p\}$, which are intentionally not included in the initial HR/LR training image sets (as showed by the auxiliary output branches of the Master Decoder in Fig. 2). These generated auxiliary vectors are subsequently used in step 3 to guide the supervised training of the Follower AE.

Second, the training procedure facilitates the simultaneous training of the Master and Follower AEs in an *alternating* manner, which may remind the training dynamics seen in GANs. In GANs, the generator and discriminator networks are trained through a competitive interaction. However, within the MR-TRAE, the relationship between the Master and Follower AEs is cooperative, where the Master AE helps in training the Follower AE.

Complexity Analysis and Implementation Aspects

Theoretically, the goal of the MR-TRAE model depicted in Fig. 2 can also be achieved by independently training p *single-resolution* TRAE models, similar to that shown in Fig. 1, each operating at a distinct scaling factor. This alternative strategy is hereafter termed as multiple single-resolution TRAE (M-SR-TRAE). However, there are at least two reasons why the MR-TRAE method is favored over the M-SR-TRAE approach.

Firstly, it is expected that multi-resolved versions of the same image at different sizes will display significant statistical correlation. The MR-TRAE model is designed to use this cross-correlation by (i) utilizing a single twinned AE block across multiple resolution scales and (ii) conducting joint training of the entire sequence of final up-sampling modules. The experimental findings presented in "Performance Results and Comparisons" section validate that using inter-image cross-correlation during the training phase effectively reduces overall training duration.

Secondly, the number of trainable parameters in the MR-TRAE, denoted as $N^{(MR-TRAE)}$, is calculated as follows:

$$N^{(MR-TRAE)} = N_{AE}^{(M)} + N_{AE}^{(F)} + \left(p \times N^{(U)}\right). \tag{16}$$

In the given equation, $N_{AE}^{(M)}$ (respectively, $N_{AE}^{(F)}$) represents the number of trainable parameters within the Master AE (respectively, the Follower AE), whereas $N^{(U)}$ denotes the number of trainable parameters for each up-sampling module.

Under the M-SR-TRAE approach, the single-resolution TRAE model shown in Fig. 1 is trained from scratch p times, each under a different scaling factor. As a result, the total number of parameters $N^{(M-SR-TRAE)}$ that need to be learned in the M-SR-TRAE approach is p times the number of trainable parameters of a single TRAE. This can be expressed as follows:

$$N^{(M-SR-TRAE)} = p \times \left(N_{AE}^{(M)} + N_{AE}^{(F)}\right) + \left(p \times N^{(U)}\right). \tag{17}$$

The complexities of the stated models are mainly determined by the complexities of their respective AEs (as further discussed in the sections on implementation aspects and the setup of implemented models in "Experimental Setup" section), a direct comparison between the formulas in (16) and (17) leads to the conclusion that the training complexity of the proposed MR-TRAE model is nearly p times lower than that of the M-SR-TRAE approach.

This conclusion is more supported by insights into implementation aspects of the MR-TRAE, as detailed in the following discussion.

On the Implemented AE Architectures Inspired by leading models for high-quality SISR as recently analyzed, for instance, in [11], Fig. 3 shows the architecture of the Master and Follower AEs as implemented.

Fundamentally, it consists of a pair of convolutional deep Encoder and Decoder networks, featuring a symmetrical architecture with mirrored hidden layers interconnected by several local shortcuts. In the context of the MR-TRAE, the auxiliary output branches shown in Fig. 3 are specifically utilized within the Master AE's implementation. On reviewing Fig. 3, the AE architecture represents four characteristics.

1. Symmetry and hourglass shape: This design ensures that the input image is systematically compressed (encoded) and subsequently expanded (decoded) as it traverses from the input to the output of the AE. Each Encoder layer's downscaling factor matches the upscaling factor of its mirrored decoder layer to maintain the architectural balance.
2. Residual-type architecture: Implemented through shortcuts between mirrored encoder-decoder layers, this aspect follows two goals: (i) facilitating direct information back-propagation to lower layers to avoid vanishing gradients during training and (ii) ensuring that input/output feature vectors of mirrored layers are of equal size to enable element-wise operations at summation nodes.
3. Pre-activation principle: Decoder layer outputs are linked to non-linear activation blocks (e.g., parametric ReLU or P-ReLU), positioned before summation nodes. This choice excludes other non-linearities (like batch normalization, pooling, or dropout) from the architecture, based on findings that they can reduce the visual quality of the resultant images.
4. Shared architecture for master and follower AEs: To justify the "twinned AEs" terminology, both Master and Follower AEs utilize the same architecture, despite processing LR/HR images of varying sizes. This can result in different sizes for their corresponding hidden layers.

Additionally, a series of convolutional layers of uniform size may be integrated between the core hidden layers of the Encoder and Decoder to improve the fidelity of information mapping from Encoder to Decoder feature spaces, using small (5×5) convolutional kernels as suggested by works like [42]. This is an optional adjustment that can refine the AE's functionality.

On the Implemented Up-sampling Modules In the MR-TRAE, each up-sampling module is implemented by using a deconvolutional layer, also known as a transposed convolutional layer. Within this model, all convolutional layers are designed to have the same spatial up-sampling factor, typically set to $us = 2$. Consequently, the total scaling factor $s_U^{(k)}$ achieved by the series of the first k deconvolutional layers equals $k \times us$, for $k = 1, ..., p$. To minimize de-blurring effects often associated with extensive zero-padding, each deconvolutional layer is fitted with small-sized (e.g., 3×3) convolutional kernels. This configuration enables each up-sampling module to scale the processed image by a factor of $\times 2$, using kernel-based convolution with a stride of one and minimal zero-padding, as detailed in "Experimental Setup" section relating to the simulation setup. The output of these

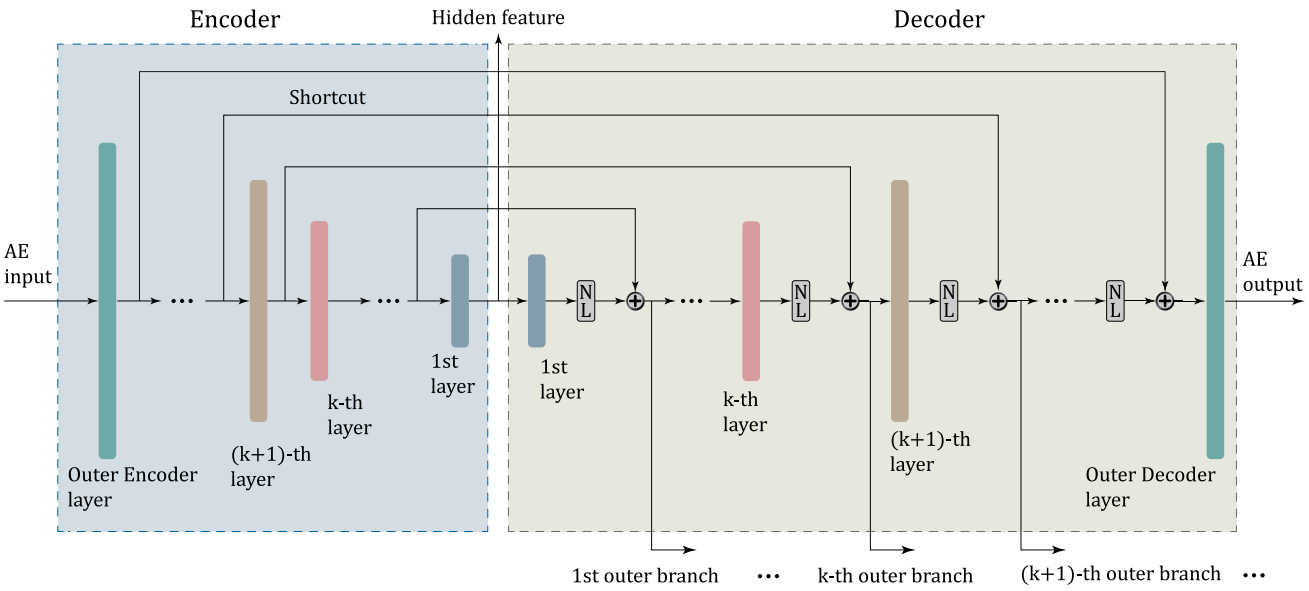


Fig. 3 The general residual-type architecture of the implemented Master and Follower convolutional AEs. The summation nodes act on a per-pixel basis. The auxiliary output branches are present only in the Master AE. Hidden layers of the same color are mirrored layers. NL, non-linearity

up-sampling modules represents the series of SR image vectors at scaling factors of us, $2 \times us$, $4 \times us$, ..., $p \times us$.

On the Implemented Inter-feature Distance Metric The distance module in Fig. 2 calculates the distance between feature vectors of the Master and Follower AEs for each pair of LR and HR input images. These feature vectors are structured to be of the same size. Any metric for measuring inter-vector distance, such as Euclidean, absolute, or cosine distances, can be applied in (12). However, a review of the works by [11], [15], and [42] suggests that the most appropriate distance metric is dependent on the specific task and application. From our numerical experiments, we have found that the Euclidean distance between inter-feature vectors generally offers satisfactory results, both in terms of the visual quality and in the accuracy of COVID vs. non-COVID classification.

On the Setting of the Trade-off Parameter γ The perceptual and content loss functions, as specified in (11), are designed to measure the image gaps under cross-related, different domains. The perceptual loss operates within the hidden feature domain for minimization that is particularly relevant when the SR images are used by non-human systems for task-oriented applications, such as image classification, as noted in [15]. Conversely, the content loss functions in the spatial or pixel domain to making its reduction beneficial for human users focused on achieving high visual quality in SR images, as highlighted in [2]. Generally, a decrease in perceptual (respectively, content) loss is associated with an improvement in classification accuracy (respectively, peak signal-to-noise ratio (PSNR) and/or structural similarity index (SSIM)) [11].

There is no guarantee of a consistent correlation between perceptual and content losses; therefore, the introduction of the γ hyper-parameter in (11) aims to balance these two foundational losses. An optimal adjustment of γ is important for achieving a desirable balance among perceptual-related and content-related performance metrics to find an optimal accuracy-vs.-PSNR-vs.-SSIM trade-off. Through a comprehensive grid search within the experimental framework described in "Experimental Setup" section, it has been determined that optimal values for the γ hyper-parameter tend to be around 0.9.

MR-TRAE Testing

In MR-TRAE, the Master AE guides the Follower AE through the training process. Therefore, once the training phase is concluded, there is no further need for either the Master AE or the HR input images. During the test phase, the MR-TRAE simplifies to include (i) the trained Follower AE; (ii) the skip connection surrounding the Follower AE, adjusted by the trained gain parameter δ; and (iii) the sequence of trained up-sampling modules. These components produce various SR versions of each LR test input image at distinct scaling factors. It enables users to choose the most fitting SR image from the generated set. In the test phase, the MR-TRAE operates on LR test images, meaning HR images are not used for testing or inference.

The trained MR-TRAE can be deployed in at least two distinct application contexts. In the first scenario, the array of multi-resolution images generated by the model can be directly examined by human observers, such as medical professionals. In the second scenario, the SR images are fed into an automated system capable of executing image classification tasks. We mention that the efficacy of the MR-TRAE across both these application scenarios is evaluated through numerical analysis in "Performance Results and Comparisons" section.

MR-TRAE Novelties and Related Cognitive Aspects

A distinctive and, to the authors' knowledge, novel aspect of the MR-TRAE is its use of multiple auxiliary outputs from the Master AE's Decoder. These outputs, which are generated at incrementally higher spatial resolutions by the inner layers, serve as reference signals for simultaneously training the Follower AE and the sequence of up-samplers. This feature eliminates the need for input images at every multi-resolution scaling factor, making the training process *semi-blind*. Moreover, the training is *domain-oblivious*, as it does not presuppose any specific knowledge about the content of the training images.

This innovation fundamentally differs from the single-resolution TRAE model illustrated in Fig. 1 and makes an advancement in the MR-TRAE. Additionally, these advancements are closely related to broader cognitive paradigms such as teacher-student (TS) learning, transfer learning (TL), knowledge distillation (KD), and hierarchical cognition (HC).

MR-TRAE Connection to the KD and TS Paradigms The knowledge distillation (KD) paradigm primarily focuses on transferring knowledge during the training phase from one model to another, typically from a more complex or resource-intensive "teacher" model to a simpler or resource-constrained "student" model, as outlined in [47]. The conventional use of KD involves teaching a student model using softer information labels distilled by a teacher model, which is usually more complex [48].

Within the MR-TRAE, we can identify that (i) the Master and Follower AEs assume the roles of teacher and student respectively and (ii) the distilled knowledge consists of the hidden features and SR images produced by the Master AE's encoder and its auxiliary output branches. From this perspective, the training steps outlined in "Semi-blind MR-TRAE Training-Loss Functions and Training Procedure"

section—namely, step 1, step 2, and step 3—play the roles of teaching, knowledge distillation, and transfer learning, respectively.

The Master and Follower AEs in Fig. 2 are designed as "twinned," which implies similar model complexities (as discussed in "Complexity Analysis and Implementation Aspects" section). Consequently, the application of the teacher-student (TS) learning model within the MR-TRAE does not address the typical goal of bridging a complexity gap between the teacher and student networks. Instead, it uses the TS paradigm to bridge the informational divide arising from using LR and HR training images of differing qualities for the Follower and Master AEs, respectively. This approach represents an unconventional application of the TS paradigm that makes the challenges associated with the quality disparity of LR/HR training images less severe.

MR-TRAE Connection to the Hierarchical Cognitive Paradigm The human cognitive system processes objects in a sequential and hierarchical manner, initially concentrating on broad features before shifting focus to more complex details [49]. Traditional single-resolution SISR models do not replicate this layered processing approach of the human brain, as they analyze input images uniformly, using convolutional layers with a singular kernel size [11]. In response to this limitation and to better align with the hierarchical processing characteristic of human cognition:

(i) The MR-TRAE uses a multi-branch architecture to present the spatial features of each HR input image across various scales. This approach enables the model to simultaneously address local details and overarching semantic aspects.
(ii) The MR-TRAE employs a hierarchically structured sequence of up-samplers to decompose the complex challenge of achieving a high up-scaling factor into a series of simpler, incremental sub-tasks, where each one targets a smaller up-scaling factor.

Within this framework, the hourglass shape of the Master AE, as shown in Fig. 3, ensures that the outputs from the Master decoder's multiple branches are organized in a hierarchical manner. Each subsequent output includes more spatial detail than its predecessor to ease a progressively refined representation of the image.

The concept of multiple auxiliary outputs, which also features in neural networks with early exits as discussed in [50], diverges in purpose and cognitive alignment from those used in the MR-TRAE model. Firstly, models with multiple early exits aim to balance the conflicting goals of achieving reliable and swift inference [50]. Unlike the auxiliary outputs in the MR-TRAE model, early exits usually incorporate local classifiers to enable quicker and potentially sufficiently accurate, local decisions. Secondly, the study in [50] associates early-exit models more closely with the cognitive principle of biological plausibility in network training, in contrast to the MR-TRAE's emphasis on teacher-student learning models and hierarchical cognition principles. However, integrating multiple auxiliary outputs within a neural network draws from a solid foundation inspired by cognitive processes in both contexts.

Experimental Setup

The numerical experiments were conducted on a PC running Windows, equipped with (i) an AMD Ryzen 9 5900X 12-Core processor at 3.7 GHz, (ii) two GeForce RTX 3070 graphics cards, and (iii) 128 GB of RAM. The Adam optimizer [47] was chosen for training purposes. The learning rate ρ during training was set to 10^{-4}, with other Adam parameters remaining at their default settings ($\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\varepsilon = 10^{-7}$). The size of the mini-batches used was $|MB| = 16$, and the training extended over 200 epochs. The up-scaling factors for the images were set at $\times 2$, $\times 4$, and $\times 8$. Detailed hyper-parameter configurations are provided in Table 2. The development of the source code

Table 2 Setting of the main simulated hyper-parameters

Parameter	Meaning/role	Setting
$\|MB\|$	Mini-batch size	16
γ	Trade-off hyper-parameter	0.9
E	Number of performed training epochs	200
ρ	Adam learning rate	10^{-4}
β_1	Adam first-moment parameter	0.9
β_2	Adam second-moment parameter	0.999
ε	Adam ε parameter	10^{-7}
$H_H \times V_H \times C$	HR training image size	$(512 \times 512 \times 1)$
$H_L \times V_L \times C$	LR test/training image size	$(64 \times 64 \times 1)$
s_U	Scaling factor	$\times 2$, $\times 4$, $\times 8$

was carried out in Python 3.10, with TensorFlow 2.10.1 utilized for additional features.

Utilized Datasets The HR images used in this study are from the publicly accessible COVIDx CT-2 A dataset [16], which includes more than 194,000 grayscale CT slices ($C = 1$) from 3700 anonymized patients. This dataset classifies the scans into three categories: COVID-19, common pneumonia, and normal controls. From this dataset, 3000 HR images were selected on a per-class basis, and their corresponding LR counterparts were created by down-sampling using a bicubic kernel, as described in (1). This process resulted in training, validation, and test sets for each class, containing 2000, 500, and 500 images, respectively. These images were pre-processed by normalizing the pixel values to a floating point range of [0, 1] and uniformly cropping the images to ensure HR and LR inputs were of the correct dimensions, as detailed in Table 2.

Implemented MR-TRAE Model The MR-TRAE operates at $\times 2$, $\times 4$, and $\times 8$ scaling factors, by using three up-sampling modules of identical architecture, each providing a $\times 2$ scale increase. The SR image output by the three up-samplers measures (128×128), (256×256), and (512×512). The uniform design of these up-samplers was specifically optimized for this multi-resolution setting, with details in Table 3. The Master and Follower AEs follow the configurations optimized and listed in Table 5 of [13], in a combined total of $20,733,953 + 8,031,746 = 28,765,699$ trainable parameters of Master-plus-Follower AEs. Before the entire MR-TRAE was fine-tuned, the Follower AE is pre-trained on 20% of the LR training images to speed up the whole training process.

In our study, the primary metrics employed to assess and compare the visual quality of the generated SR images include the peak signal-to-noise ratio (PSNR) and the structural similarity index (SSIM) [4]. Briefly:

Table 3 Common architecture of the implemented convolutional up-sampling modules. Each module provides a scaling factor of $\times 2$. $Conv2D(F, K)$ indicates a convolutional layer with F filters and a kernel size of $(K \times K)$. $D2S(B)$ denotes a depth-to-space layer that rearranges data from the depth dimension into blocks of size $(B \times B)$, so as to increase the resulting spatial resolution. $\#TP$, number of trainable parameters

Layer	Up-sampler ($\times 2$)
Layer #1	$Conv2D(128, 5)$
Layer #2	$Conv2D(64, 3)$
Layer #3	$D2S(2)$
Layer #4	$Conv2D(64, 3)$
Layer #5	$Conv2D(1, 3)$
$\#TP$	86,977

(i) The PSNR index measures the rendered pixel-level accuracy compared to the reference image. It is defined as follows:

$$\text{PSNR (dB)} = 10 \times \log_{10}\left(\frac{\text{MAX}_I^2}{\text{MSE}}\right), \tag{18}$$

where MAX_I is the pixel peak-value (i.e., it equates to 255 when 8 bits per pixel are used), while MSE is the mean square error between the reference image and its recovered super-resolved version.

(ii) The SSIM index measures the perceived change in structural information and texture between the reference and recovered images. It provides a more perceptually relevant measure of image quality by comparing local patterns of pixel intensities, which have been normalized with respect to luminance and contrast. Hence, the evaluation of the SSIM index over two equal-size spatial windows x and y extracted from two corresponding locations of the reference and recovered images is as follows [4]:

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}, \tag{19}$$

where (a) μ_x and μ_y are the average pixel values of the images x and y, while σ_x^2 and σ_y^2 are the corresponding variances; (b) σ_{xy} is the cross-covariance between x and y; and (c) C_1 and C_2 are small positive constants that are introduced to avoid divisions by zero. Typically, we have [4] $C_1 = (k_1 L)^2$ and $C_2 = (k_2 L)^2$, where L is the pixel range (255 for 8-bit coded images), with $k_1 = 0.01$ and $k_2 = 0.03$. The SSIM index assumes values over the interval: $[-1, 1]$, with SSIM $= 1$ denoting a perfect similarity.

Before proceeding, we point out that, in our framework, ground-truth super-resolved images are not available by design. Hence, as typically done in the literature (see, for example, [51]), we generated super-resolved reference images of sizes: (128×128) and (256×256) by down-sampling the available (512×512) HR images through bicubic kernel.

Performance Results and Comparisons

This section aims to evaluate the performance of the MR-TRAE during training and testing phases in comparison to two baseline models: the U-Net in [14] and the M-SR-TRAE benchmark described in "Complexity Analysis and Implementation Aspects" section.

The rationale behind choosing these models for comparison begins with the U-Net architecture's original devel-

opment for biomedical image segmentation, where it has shown remarkable performance robustness across various applications [4]. Its inclusion as a primary benchmark stems from two key points. Firstly, the U-Net model is specifically designed for handling medical imagery, such as CT scans, making it an appropriate reference for our purposes. Secondly, the layered CNN-based structure of U-Net shares similarities with the architecture of our MR-TRAE model, providing a basis for meaningful performance comparison.

The M-SR-TRAE model is considered a secondary baseline primarily because the outcomes achieved by individual single-resolution models could serve as an upper-bound performance benchmark for their multi-resolution counterparts. The idea is that training a single up-sampler at a time is likely to be more effective than simultaneously training multiple up-samplers, as it can be noticed by comparing the structures in Figs. 1 and 2. Consequently, a trade-off between the M-SR-TRAE and MR-TRAE models is expected to involve a balance between training duration and test performance. The empirical findings shared in "Performance Results and Comparisons" section will confirm this hypothesis and provide valuable insights into the practical trade-offs involved.

Table 4 provides a basis for comparing the U-Net and MR-TRAE performance in terms of PSNR and SSIM. The companion Fig. 4 allows a visual comparison of the SR images produced by these models. For comparison purposes, the reference ground-truth HR image and the corresponding LR one are reported in the first row of Fig. 4.

Table 4 U-Net vs. MR-TRAE vs. super-resolution GAN ($SRGAN$) comparative performance. Test accuracy (ACC) is evaluated by using three pre-trained GoogLeNets as benchmark binary classifiers. Each classifier works on SR input images that are, in turn, generated by the implemented TRAE, MR-TRAE, U-Net, and SRGAN models at scaling factors $\times 2$, $\times 4$, and $\times 8$. The ideal baseline (IB), in the last row, refers to the model in which the (512×512) ground-truth HR images are directly used as inputs to the corresponding benchmark GoogLeNet classifier

Model	ACC (%)	PSNR (dB)	SSIM
TRAE @$\times 2$	95.52	29.30	0.8910
TRAE @$\times 4$	91.28	25.48	0.8716
TRAE @$\times 8$	88.63	22.25	0.7110
MR-TRAE @$\times 2$	95.47	28.50	0.8898
MR-TRAE @$\times 4$	91.24	25.18	0.8714
MR-TRAE @$\times 8$	88.61	22.10	0.7100
U-Net @$\times 2$	94.01	27.74	0.8678
U-Net @$\times 4$	89.61	24.94	0.8525
U-Net @$\times 8$	88.60	22.51	0.7159
SRGAN @$\times 2$	94.50	27.82	0.8838
SRGAN @$\times 4$	90.50	24.55	0.8702
SRGAN @$\times 8$	86.50	20.32	0.7054
IB	95.90	∞	1.0000

A column-wise comparison of the entries in the first nine rows of Table 4 shows that the U-Net underperforms compared to the MR-TRAE, particularly at smaller SR image dimensions, i.e., at image sizes of (128×128) and (256×256). Furthermore, even at the maximum considered image size of (512×512), the performance gap between the MR-TRAE and the U-Net remains quite limited (see the sixth and ninth rows of Table 4).

A visual comparison of the SR and ground-truth HR images in Fig. 4 further corroborates these insights and allows us to conclude that

1. The U-Net model tends to introduce checkerboard artifacts in the rendered SR images, especially at higher up-scaling factors. Such artifacts are absent (or, at least, not noticeable) in the corresponding SR images generated by the MR-TRAE model.
2. A column-wise comparison of the reported images indicates that the visual quality of the images rendered by the MR-TRAE model improves as the scaling factors increase, while this performance trend is not present in the corresponding SR images generated by the U-Net model.
3. The MR-TRAE capability to capture and reproduce fine spatial details is seen when comparing the ground-truth HR image in Fig. 4 against the corresponding SR ones in the last row of Fig. 4. In this regard, we note that the inset of the ground-truth HR image in the first row of Fig. 4 exhibits snowflake-like patterns representing high-frequency spatial details which are (i) missing in the corresponding LR input image, (ii) no longer noticeable due to pixelation in the U-Net's SR image reported in the first column of the last row in Fig. 4), and (iii) still detectable in the corresponding SR image rendered by the MR-TRAE (see the inset of the second column in the last row of Fig. 4).

All in all, these observations provide (a first) support to the conclusion that the hierarchical feature extraction and knowledge distillation performed by the multi-branch Master AE of Fig. 2 during training phase help the trained Follower AE to accurately render finer spatial details in the corresponding inference phase.

Multi-resolution vs. Multiple Single-resolutions

In principle, *multiple single-resolution* TRAE models, similar to those described in [13], could be *independently* trained, each targeting a *distinct* resolution factor. This approach would result in *several* trained models, with each capable of producing good quality images at a *single* scaling factor.

This observation leads to a key question: why should we opt for the MR-TRAE model over using several single-resolution models like the one of Fig. 1?

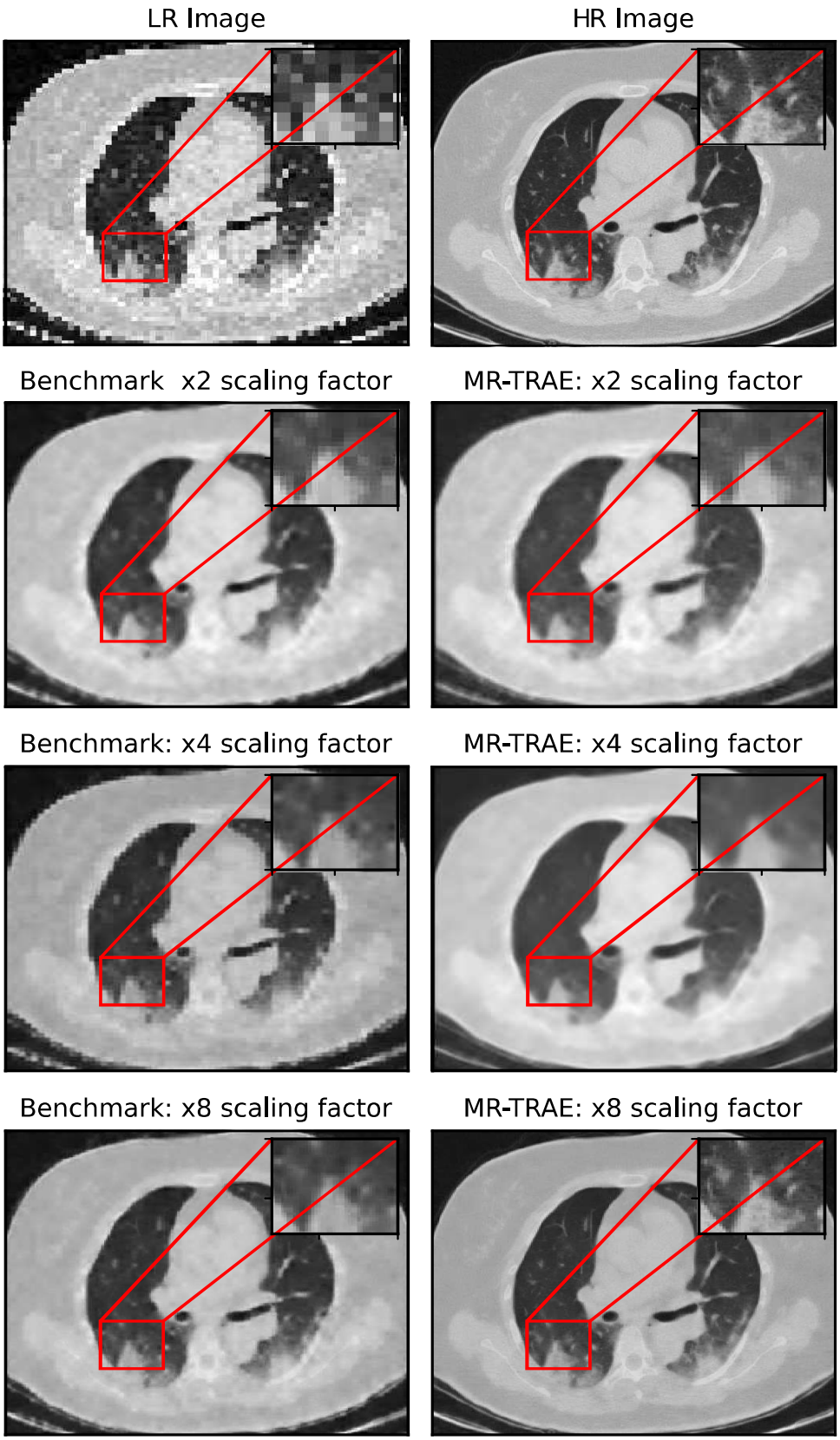


Fig. 4 Instances of SR images rendered by the implemented MR-TRAE and U-Net models at scaling factors $\times 2$, $\times 4$, and $\times 8$. The first row presents the basic 64×64 LR input and the corresponding ground-truth 512×512 HR images. Insets highlight fine spatial details to emphasize the differences in the visual quality of the SR images rendered by the implemented MR-TRAE and U-Net models

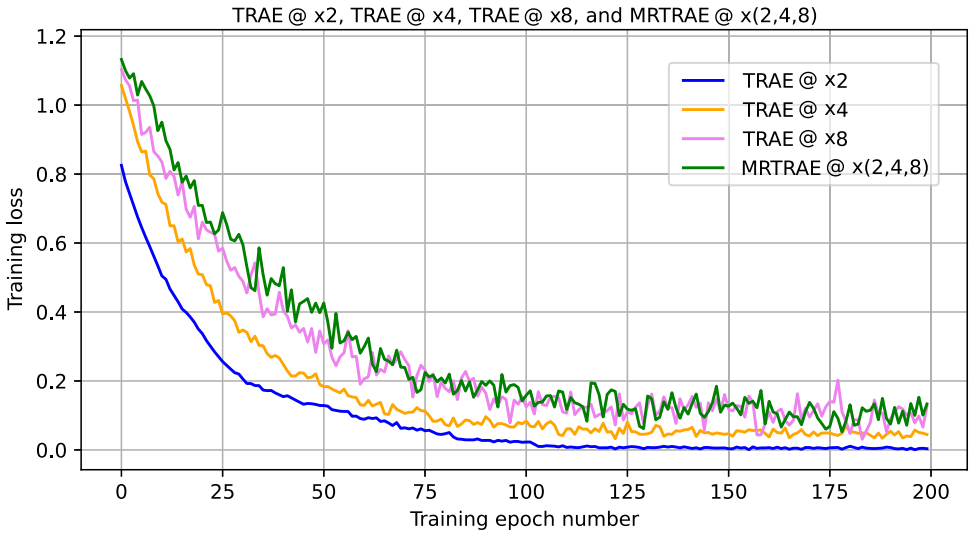


Fig. 5 Simulated training loss curves under the proposed MR-TRAE model at multi-resolutions $\times(2, 4, 8)$ (green plot) and the corresponding benchmark single-resolution TRAE models at scaling factors $\times 2$ (blue plot), $\times 4$ (yellow plot), and $\times 8$ (magenta plot)

To address this question, we have carried out simulations to assess both the training and testing performance of the MR-TRAE, as well as the corresponding ones of three single-resolution TRAE models, with scaling factors of $\times 2$, $\times 4$, and $\times 8$. For a fair comparison, these benchmark TRAE models are the ones designed in [13], and henceforth, they will be referred to as TRAE @$\times 2$, TRAE @$\times 4$, and TRAE @$\times 8$. The resulting data from the carried out simulations are presented in Fig. 5 and in the first six rows of Table 4 for the training and test phases, respectively. Additionally, Table 5 compares the associated implementation complexities in terms of the number of trainable parameters and measured training times.

As already pointed out in "MR-TRAE Testing" section, the test accuracy presented in Table 4 refers to the binary classification of COVID/non-COVID CT images processed by the TRAE and MR-TRAE models at scaling factors of $\times 2$, $\times 4$, and $\times 8$. For each of them, a pre-trained GoogLeNet serves as a benchmark binary classifier. Specifically, the implemented GoogLeNet receives in input the SR images rendered by the TRAE/MR-TRAE models at the given scaling factor and, then, classifies them as COVID/non-COVID ones.

In the simulations, three separate GoogLeNets are trained to cope with SR input images of sizes of (128×128), (256×256), and (512×512). To align with the considered testing frameworks, we introduce the following taxonomy: (i) MR-TRAE @$\times(2, 4, 8)$ denotes an MR-TRAE model that simultaneously trains three up-samplers at scaling factors $\times 2$, $\times 4$, and $\times 8$ (see the last column of Table 5 and the green plot of Fig. 5), while (ii) MR-TRAE @$\times 2$, MR-TRAE @$\times 4$, and MR-TRAE @$\times 8$ are used to refer to the test performance of the (aforementioned) MR-TRAE @$\times(2, 4, 8)$ model when evaluated at the scaling factors of $\times 2$, $\times 4$, and $\times 8$ in the testing phase (see the forth, fifth, and sixth rows of Table 4).

A comparison of the numerical results presented in the first six rows of Table 4 and in Table 5 allows us to acquire three main insights about the trade-off between using the multi-resolution approach versus the multiple single-resolution ones.

Firstly, the data in Table 4 indicate that the test performance metrics for the MR-TRAE are nearly as good as those of the corresponding single-resolution TRAE models, with only marginal differences, even at $\times 8$ scaling factor.

Secondly, the first row of Table 5 shows the total number of trainable parameters for the implemented MR-TRAE is approximately 0.35 times the aggregate number of the trainable parameters of the corresponding single-resolution TRAE models. This results, in turn, in a reduction of the overall model complexity of about 65%.

Third, the last row of Table 5 shows that the measured wall-clock times for training the three single-resolution TRAE @$\times 2$, TRAE @$\times 4$, and TRAE @$\times 8$ models are 6:10, 7:15, and 8:10 (in hours and minutes) respectively. This results in a total training time of 1295 min (21:35 in hours and minutes), which is approximately 2.46 times greater than the 525 min

Table 5 Numbers of the overall trainable model parameters and measured training times. $\#TP$, number of trainable model parameters; TT, training time (hour:minute)

Parameter	**TRAE @$\times 2$**	**TRAE @$\times 4$**	**TRAE @$\times 8$**	**MR-TRAE @$\times(2, 4, 8)$**
#TP	30,189,972	29,431,380	28,818,068	29,026,630
TT	6:10	7:15	8:10	8:45

(8:45 in hours and minutes) needed to train the corresponding single MR-TRAE @$\times(2, 4, 8)$ model. This finding is corroborated by the convergence times, in terms of the number of training epochs of the corresponding training loss curves presented in Fig. 5.

In summary, the reduction in training time achieved by using a single MR-TRAE model in place of multiple single-resolution TRAE models is offset by a minor reduction in test accuracy. This trade-off confirms the effectiveness of the MR-TRAE in using the cross-correlation found among different resolutions of the same LR input image.

Finally, we point out that a comparison of the data in Table 4 with the rendered SR images in Fig. 5 shows a discrepancy between performance metrics and corresponding visual quality. In fact, while the performance metrics in Table 4 tend to decrease with increasing scaling factors, the visual quality of the images in Fig. 5 appears to improve for higher scaling factors. This confirms, indeed, that the performance sensitivity of human observers and automated machines do not always align [12].

Comparison with GAN-Based Baselines

In this section, we test and compare the performance of the MR-TRAE model against one of the SISR GAN-based models introduced in [15], generally referred to as super-resolution GAN (SRGAN). Specifically, SRGAN uses a GAN, with its generator using a ResNet architecture featuring skip connections, as detailed in Fig. 4 of [15]. Furthermore, the discriminator network of the SRGAN model is composed of eight convolutional layers, adopting a design similar to the VGG network. The discriminator uses convolutional kernels of size (3×3), with the number of kernels doubling progressively from the input layer, starting with 64 kernels, to the output layer with 512 kernels.

The resulting SRGAN, with a total of 43,487,690 trainable parameters, optimizes a perceptual loss function that combines content and adversarial losses, as detailed in Section 2 of [15]. This enables SRGAN to generate SR images free from noticeable hallucination effects, a claim supported by both objective and subjective assessments in [15]. Unlike the multi-resolution approach featuring the proposed MR-TRAE model, SRGAN operates in a single-resolution fashion, so that a SRGAN model must be independently trained for each considered scaling factor.

The framework adopted for the SRGAN training is sketched in Fig. 6. To evaluate the effectiveness of the trained SRGAN and carry out a fair performance comparison with the MR-TRAE, we use the testing framework outlined in Fig. 7.

In this setup, the final component is a pre-trained GoogLeNet serving as a binary classifier. For benchmarking purposes, we also numerically checked the test accuracy

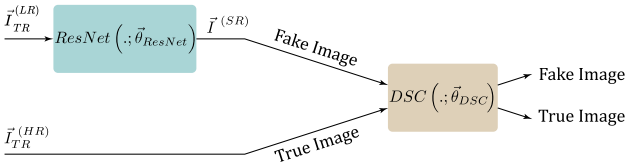


Fig. 6 The simulated single-resolution GAN-based baseline [15]

of the implemented GoogLeNet in an ideal scenario where GoogLeNet directly classifies ground-truth HR input images. This scenario, referred to as ideal baseline (IB), provides an upper bound on the classification performance achievable when using SR images as inputs for testing.

The obtained SRGAN performance presented in the bottom part of Table 4 exhibits, indeed, a degrading trend as the scaling factor increases. Specifically, SRGAN performance is slightly lower than the corresponding one of the MR-TRAE in terms of classification accuracy, PSNR, and SSIM metrics. This conclusion is further corroborated by the fact that the classification accuracy achieved by the MR-TRAE model closely approaches the performance of the ideal baseline (see the last row of Table 4).

Figure 8 shows a visual comparison between (i) an SR image rendered by the simulated SRGAN at $\times 8$ scaling factor (see Fig. 8b), (ii) the corresponding SR image generated by the MR-TRAE (see Fig. 8c), and (iii) the reference ground-truth HR image of the size (512×512) (see Fig. 8a). The visual comparison of these images gives a first evidence of the performance superiority of the MR-TRAE architecture over SRGAN one. This is due to the fact that MR-TRAE is capable of generating SR images that are free from noticeable hallucination effects.

To reinforce this (visual comparison-based) conclusion with objective and quantitative metrics, Fig. 8d displays the per-pixel absolute difference map between the ground-truth HR image in Fig. 8a and the corresponding SR image produced by the SRGAN baseline in Fig. 8b. In a similar way, Fig. 8e presents the per-pixel absolute difference map between the ground-truth HR image in Fig. 8a and the SR image generated by the MR-TRAE model in Fig. 8c. In the presented difference maps, white and black dots represent the minimum (zero) and maximum (one) per-pixel absolute differences, respectively.

In both different maps, darker dots are more noticeable in areas rich in spatial details in the ground-truth HR image.

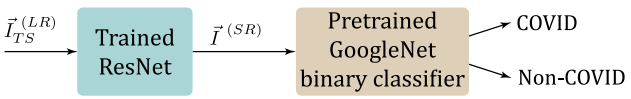
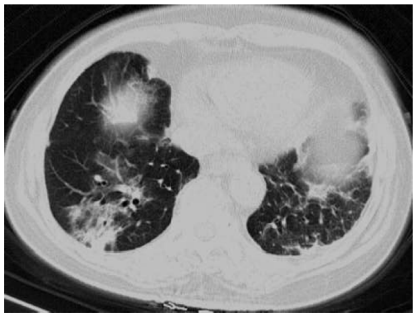
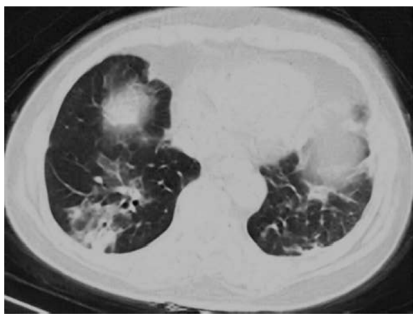


Fig. 7 Simulated benchmark scheme for testing and comparing the performance of the trained MR-TRAE model against the GAN-based baseline

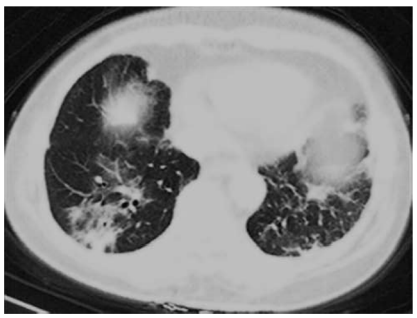
Fig. 8 Visual comparison between super-resolved images and corresponding difference maps



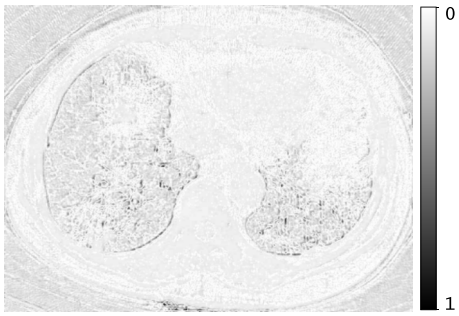
(a) Ground-truth (512 × 512) HR image.



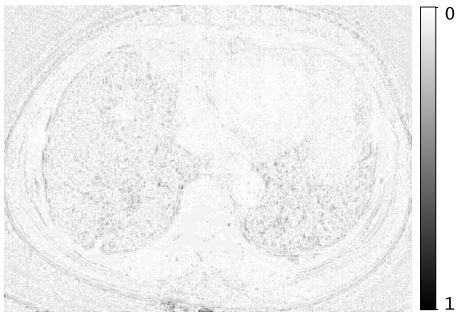
(b) SR image generated by SRGAN at scaling factor ×8.



(c) SR image generated by MR-TRAE at scaling factor ×8.



(d) Differential map between the original image in Figure 8a and the rendered SRGAN image in Figure 8b.



(e) Differential map between the original image in Figure 8a and the rendered MR-TRAE image in Figure 8c.

On average, both the spatial density and the intensity of the darker dots in Fig. 8d are greater than those in Fig. 8e. This indeed provides additional (objective) evidence of the superior SR performance achievable by the MR-TRAE model.

Conclusion and Hints for Future Research

In this study, we have developed and evaluated the MR-TRAE, a DL architecture for semi-blind training in SISR. This MR-TRAE framework broadens the recently introduced TRAE concept from [13], traditionally applied to single-resolution image processing, to include multi-resolution capabilities. The main features of the MR-TRAE include (i) the incorporation of a series of trainable CNN-based upsamplers into the foundational TRAE structure and (ii) the formulation of a specialized loss function for their integrated semi-blind training. These features of the MR-TRAE model are aligned with cognitive learning concepts such as knowledge distillation, the teacher-student learning paradigm, and hierarchical cognition. The comparative evaluations presented herein confirm that the MR-TRAE model efficiently compresses the overall training duration without significantly affecting test accuracy or the visual integrity of the SR images.

These findings pave the way for at least two directions of research.

First, to carry out fair comparisons with previous state-of-the-art models, we tested the MR-TRAE model on open-access grayscale CT scans. However, the MR-TRAE and its associated training loss function could be used for SR across natural and color images. This can be achieved by incorporating appropriate color channels into the AEs and up-samplers shown in Fig. 2. Therefore, evaluation of the MR-TRAE performance with respect to different color spaces presents a promising and practical direction for further research.

A second line for research stems from the observation that the MR-TRAE shows lower model complexity and shorter training duration compared to using several single-resolution models across various scaling factors. Consequently, another research direction could explore the distributed training of the MR-TRAE model on multi-tier edge computing infrastructures. This approach aims to enable resource-constrained end-user devices to use wireless connectivity to proximate edge-based proxy servers for executing distributed or federated training of the MR-TRAE network.

Author Contribution Conceptualization: Enzo Baccarelli. Methodology: Enzo Baccarelli. Writing—original draft preparation: Alireza Momenzadeh and Enzo Baccarelli. Writing—review and editing: Alireza Momenzadeh. Formal analysis: Alireza Momenzadeh. Supervision: Enzo Baccarelli. Software: Michele Scarpiniti and Sima Sarv Ahrabi. Data validation: Alireza Momenzadeh and Enzo Baccarelli. Visualization: Alireza Momenzadeh. Data curation: Alireza Momenzadeh.

Funding Open access funding provided by Università degli Studi di Roma La Sapienza within the CRUI-CARE Agreement. This work is partially supported by the research collaboration with the Institute of Informatics and Telematics (IIT-CNR) on the theme: "Design and software development of Fog Computing platforms for the support of Deep Learning algorithms for the real-time image analysis." The work has been also supported by the projects: "Flying Fog (FF): when the Fog learns to fly" funded by the Sapienza University of Rome Bando 2022 and 2023.

Data Availability The data set used in this work (COVIDx CT-2 A) can be accessed via https://www.kaggle.com/datasets/hgunraj/covidxct

Declarations

Ethical Approval This article does not contain any studies with human participants or animals performed by any of the authors.

Consent to Participate Not applicable.

Competing Interests The authors declare no competing interests.

References

1. Chen H, He X, et al. Real-world single image super-resolution: a brief review. Inf Fusion. 2022;79:124–45. https://doi.org/10.1016/j.inffus.2021.09.005.
2. Chauhan K, Patel SN, et al. Deep learning-based single-image super-resolution: a comprehensive review. IEEE Access. 2023;11:21811–30. https://doi.org/10.1109/ACCESS.2023.3251396.
3. Villar-Corrales A, Schirrmacher F, Riess C. Deep learning architectural designs for super-resolution of noisy images. In: ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). ICASSP'21. Toronto: IEEE; 2021. pp. 1635–39. https://doi.org/10.1109/ICASSP39728.2021.9414733.
4. Lepcha DC, Goyal B, et al. Image super-resolution: a comprehensive review, recent trends, challenges and applications. Inf Fusion. 2023;91:230–60. https://doi.org/10.1016/j.inffus.2022.10.007.
5. Dabov K, Foi A, et al. Image denoising by sparse 3-D transform-domain collaborative filtering. IEEE Trans Image Process. 2007;16(8):2080–95. https://doi.org/10.1109/TIP.2007.901238.
6. Chen H, He X, et al. Self-supervised cycle-consistent learning for scale-arbitrary real-world single image super-resolution. Expert Syst Appl. 2023;212:118657. https://doi.org/10.1016/j.eswa.2022.118657.
7. Wang Z, Chen J, Hoi SCH. Deep learning for image super-resolution: a survey. IEEE Trans Pattern Anal Mach Intell. 2021;43(10):3365–87. https://doi.org/10.1109/TPAMI.2020.2982166.
8. Sarv Ahrabi S, Momenzadeh A, Baccarelli E, Scarpiniti M, Piazzo L. How much BiGAN and CycleGAN-learned hidden features are effective for COVID-19 detection from CT images? A comparative study. J Supercomput. 2023;79(3):2850–81. https://doi.org/10.1007/s11227-022-04775-y.
9. Chou PA, Schaar M (Eds). Multimedia over IP and wireless networks: compression, networks: compression, networking, and systems, 1st edn. California: Academic Press; 2007. https://doi.org/10.1016/B978-0-12-088480-3.X5000-0
10. Chen Y, Zheng Q, Chen J. Double paths network with residual information distillation for improving lung CT image super resolution. Biomed Signal Process Control. 2022;73:103412. https://doi.org/10.1016/j.bspc.2021.103412.
11. Li J, Fang F, et al. Multi-scale residual network for image super-resolution. In: 15th European Conference on Computer Vision (ECCV). ECCV'18. Munich; Springer: 2018. pp. 517–32. https://doi.org/10.1007/978-3-030-01237-3_32.
12. Liu A, Liu Y, et al. Blind image super-resolution: a survey and beyond. IEEE Trans Pattern Anal Mach Intell. 2023;45(5):5461–80. https://doi.org/10.1109/TPAMI.2022.3203009.
13. Baccarelli E, Scarpiniti M, Momenzadeh A. Twinned residual auto-encoder (TRAE)-A new DL architecture for denoising super-resolution and task-aware feature learning from COVID-19 CT images. Expert Syst Appl. 2023;225:120104. https://doi.org/10.1016/j.eswa.2023.120104.
14. Ronneberger O, Fischer P, Brox T. U-Net: convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-assisted Intervention–MICCAI 2015. MICCAI'15. Munich: Springer; 2015. pp. 234–41. https://doi.org/10.1007/978-3-319-24574-4_28.

15. Ledig C, Theis L, et al. Photo-realistic single image super-resolution using a generative adversarial network. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017). Honolulu: IEEE; 2017. pp. 105–14. https://doi.org/10.1109/CVPR.2017.19.
16. Gunraj H, Sabri A, et al. COVID-Net CT-2: enhanced deep neural networks for detection of COVID-19 from chest CT images through bigger, more diverse learning. Front Med. 2022;8:729287. https://doi.org/10.3389/fmed.2021.729287.
17. Dong C, Loy CC, et al. Image super-resolution using deep convolutional networks. IEEE Trans Pattern Anal Mach Intell. 2016;38(2):295–307. https://doi.org/10.1109/TPAMI.2015.2439281.
18. Lim B, Son S, et al. Enhanced deep residual networks for single image super-resolution. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). CVPRW'17. Honolulu: IEEE; 2017. pp. 1132–40. https://doi.org/10.1109/CVPRW.2017.151.
19. Zhang Y, Tian Y, et al. Residual dense network for image super-resolution. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). CVPR'18. Salt Lake City: IEEE; 2018. pp. 2472–81. https://doi.org/10.1109/CVPR.2018.00262.
20. Huang G, Liu Z, et al. Densely connected convolutional networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). CVPR'17. Honolulu: IEEE; 2017. pp. 4700–8. https://doi.org/10.1109/CVPR.2017.243.
21. Liu J, Zhang W, et al. Residual feature aggregation network for image super-resolution. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). CVPR'20. Seattle: IEEE; 2020. pp. 2356–65. https://doi.org/10.1109/CVPR42600.2020.00243.
22. Lu L, Li W, et al. MASA-SR: matching acceleration and spatial adaptation for reference-based image super-resolution. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). CVPR'21. Nashville: IEEE; 2021. pp. 6364–73. https://doi.org/10.1109/CVPR46437.2021.00630.
23. Huang Y, Li J, et al. Interpretable detail-fidelity attention network for single image super-resolution. IEEE Trans Image Process. 2021;30:2325–39. https://doi.org/10.1109/TIP.2021.3050856.
24. Wang L, Wang Y, et al. Unsupervised degradation representation learning for blind super-resolution. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). CVPR'21. Nashville: IEEE; 2021. pp. 10576–85. https://doi.org/10.1109/CVPR46437.2021.01044.
25. Zhang Y, Li K, et al. Image super-resolution using very deep residual channel attention networks. In: European Conference on Computer Vision (ECCV). ECCV'2018. Munich: Springer; 2018. pp. 294–310. https://doi.org/10.1007/978-3-030-01234-2_18.
26. Dai T, Cai J, et al. Second-order attention network for single image super-resolution. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). CVPR'19. Long Beach: IEEE; 2019. pp. 11057–66. https://doi.org/10.1109/CVPR.2019.01132.
27. Niu B, et al. Single image super-resolution via a holistic attention network. In: Computer Vision–ECCV 2020. ECCV'20. Glasgow: Springer; 2020. pp. 191–207. https://doi.org/10.1007/978-3-030-58610-2_12.
28. Wang Z, Lu Y, et al. Single image super-resolution with attention-based densely connected module. Neurocomputing. 2021;453:876–84. https://doi.org/10.1016/j.neucom.2020.08.070.
29. Liu H, Cao F, et al. Lightweight multi-scale residual networks with attention for image super-resolution. knowlBased Syst. 2020;203:106103. https://doi.org/10.1016/j.knosys.2020.106103.
30. Liu H, Cao F. Improved dual-scale residual network for image super-resolution. Neural Netw. 2020;132:84–95. https://doi.org/10.1016/j.neunet.2020.08.008.
31. Song X, Liu W, et al. Image super-resolution with multi-scale fractal residual attention network. Comput Graph. 2023;113:21–31. https://doi.org/10.1016/j.cag.2023.04.007.
32. Hu Y, Huang Y, Zhang K. Multi-scale information distillation network for efficient image super-resolution. knowlBased Syst. 2023;275:110718. https://doi.org/10.1016/j.knosys.2023.110718.
33. Ye S, Zhao S, et al. Single-image super-resolution challenges: a brief review. Electronics. 2023;12(13):2975. https://doi.org/10.3390/electronics12132975.
34. Wang L, Dong X, et al. Exploring sparsity in image super-resolution for efficient inference. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). CVPR'21. Nashville: IEEE; 2021. pp. 4915–24. https://doi.org/10.1109/CVPR46437.2021.00488.
35. Tian C, Xu Y, et al. Coarse-to-fine CNN for image super-resolution. IEEE Trans Multimedia. 2021;23:1489–502. https://doi.org/10.1109/TMM.2020.2999182.
36. Tian C, Xu Y, et al. Asymmetric CNN for image superresolution. IEEE Trans Syst Man Cybern Syst. 2022;52(6):3718–30. https://doi.org/10.1109/TSMC.2021.3069265.
37. Lu Z, Li J, et al. Transformer for single image super-resolution. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). CVPRW'22. New Orleans: IEEE; 2022. pp. 456–65. https://doi.org/10.1109/CVPRW56347.2022.00061.
38. Tran T-H, Berberich J, Simon S. 3DVSR: 3D EPI volume-based approach for angular and spatial light field image super-resolution. Signal Process. 2022;192:108373. https://doi.org/10.1016/j.sigpro.2021.108373.
39. Liu Y, Jia Q, et al. Cross-SRN: structure-preserving super-resolution network with cross convolution. IEEE Trans Circuits Syst Video Technol. 2022;32(8):4927–39. https://doi.org/10.1109/TCSVT.2021.3138431.
40. Aakerberg A, Johansen AS, et al. Semantic segmentation guided real-world super-resolution. In: 2022 IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW). WACVW'22. Waikoloa: IEEE; 2022. pp. 449–58. https://doi.org/10.1109/WACVW54805.2022.00051.
41. Kong F, Li M, et al. Residual local feature network for efficient super-resolution. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). CVPRW'22. New Orleans: IEEE; 2022. pp. 765–75. https://doi.org/10.1109/CVPRW56347.2022.00092.
42. Yang A, Li L. Non-linear perceptual multi-scale network for single image super-resolution. Neural Netw. 2022;152:201–11. https://doi.org/10.1016/j.neunet.2022.04.020.
43. Bhatele KR, Jha A, et al. COVID-19 detection: a systematic review of machine and deep learning-based approaches utilizing chest X-rays and CT scans. Cognit Comput. 2022. https://doi.org/10.1007/s12559-022-10076-6.
44. Goel T, Murugan R, et al. Automatic screening of COVID-19 using an optimized generative adversarial network. Cognit Comput. 2021. https://doi.org/10.1007/s12559-020-09785-7.
45. Sun L, Liu Z, et al. Lightweight image super-resolution via weighted multi-scale residual network. IEEE/CAA J Autom Sinica. 2021;8(7):1271–80. https://doi.org/10.1109/JAS.2021.1004009.
46. Wang Y, Shao Z, et al. Remote sensing image super-resolution via multiscale enhancement network. IEEE Geosci Remote Sens Lett. 2023;20:1–5. https://doi.org/10.1109/LGRS.2023.3248069.

47. Goodfellow I, Bengio Y, Courville A. Deep learning, 1st edn. Cambridge: The MIT Press; 2016. https://mitpress.mit.edu/9780262035613/deep-learning/.
48. Li W, Du Z, et al. Hierarchical feature aggregation network for deep image compression. In: 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). ICASSP'22. Singapore: IEEE; 2022. pp. 1875–9. https://doi.org/10.1109/ICASSP43922.2022.9746628.
49. Yang D, Du Y, et al. Image semantic segmentation with hierarchical feature fusion based on deep neural network. Connect Sci. 2022;34(1):1772–84. https://doi.org/10.1080/09540091.2022.2082384.
50. Scardapane S, Scarpiniti M, et al. Why should we add early exits to neural networks? Cognit Comput. 2020. https://doi.org/10.1007/s12559-020-09734-4.
51. Chen W, Ma Y, et al. Hierarchical generative adversarial networks for single image super-resolution. In: IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). WACV'21. Waikoloa: IEEE; 2021. pp. 355–64. https://doi.org/10.1109/WACV48630.2021.00040.