Attention-based Multi-Reference Learning for Image Super-Resolution

Marco Pesavento Marco Volino Adrian Hilton
Centre for Vision, Speech and Signal Processing
University of Surrey, UK
{m.pesavento,m.volino,a.hilton}@surrey.ac.uk

Abstract

This paper proposes a novel Attention-based Multi-Reference Super-resolution network (AMRSR) that, given a low-resolution image, learns to adaptively transfer the most similar texture from multiple reference images to the super-resolution output whilst maintaining spatial coherence. The use of multiple reference images together with attention-based sampling is demonstrated to achieve significantly improved performance over state-of-the-art reference super-resolution approaches on multiple benchmark datasets. Reference super-resolution approaches have recently been proposed to overcome the ill-posed problem of image super-resolution by providing additional information from a high-resolution reference image. Multi-reference super-resolution extends this approach by providing a more diverse pool of image features to overcome the inherent information deficit whilst maintaining memory efficiency. A novel hierarchical attention-based sampling approach is introduced to learn the similarity between low-resolution image features and multiple reference images based on a perceptual loss. Ablation demonstrates the contribution of both multi-reference and hierarchical attention-based sampling to overall performance. Perceptual and quantitative ground-truth evaluation demonstrates significant improvement in performance even when the reference images deviate significantly from the target image. The project website can be found at https://marcopesavento.github.io/AMRSR/

1. Introduction

Image super-resolution (SR) aims to estimate a perceptually plausible high-resolution (HR) image from a low-resolution (LR) input image [38]. This problem is ill-posed due to the inherent information deficit between LR and HR images. Classic super-resolution image processing [24] and deep learning based approaches [37] result in visual artefacts for large up-scaling factors (4×). To overcome this limitation, recent research has introduced the sub-problem of reference image super-resolution (RefSR) [6, 41, 46]. Given an input LR image and a similar HR reference image, RefSR approaches estimate a SR image. Reference super-resolution with a single reference image has been demonstrated to improve performances over general SR methods achieving large up-scaling with reduced visual artefacts.

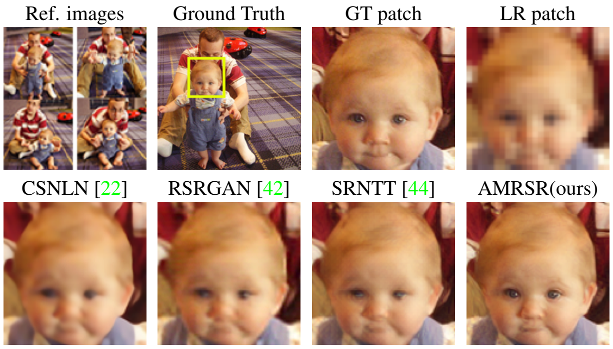

Figure 1: The proposed network exploits $N_M = 4$ reference images (top left) to super-resolve the LR input (top right). Its SR output has the best visual quality compared to other state-of-the-art methods.

In this paper we generalise reference super-resolution to use multiple reference images giving a pool of image features and propose a novel attention-based sampling approach to learn the perceptual similarity between reference features and the LR input. The proposed attention-based multiple-reference super-resolution network (AMRSR) is designed to allow multiple HR reference images by introducing a hierarchical attention-based mapping of LR input feature subvectors into HR reference feature vectors, focusing the learning attention on the LR input. This allows training with multiple HR reference images which would not be possible with a naive extension of existing single-reference super-resolution methods without a significant increase in memory footprint. Figure 1 qualitatively illustrates the performance of the proposed AMRSR approach against state-of-the-art single-image super-resolution (CSNLN [22], RSRGAN [42]) and RefSR (SRNTT [44]) approaches. Given N_M reference images, AMRSR produces a 4× SR image which is perceptually plausible and has a similar level of detail to the ground-truth HR image. The primary contributions of the AMRSR approach presented in this paper are:

- Generalisation of single reference super-resolution to multiple reference images whilst improving memory efficiency thanks to a part-based mechanism.
- Hierarchical attention-based adaptive sampling for perceptual similarity learning between low-resolution image features and multiple HR reference images.
- Improved quantitative and perceptual performance for image super-resolution compared with state-of-the-art single-image RefSR.

AMRSR is applied to both image and 3D model texture map SR where multiple HR reference images are available. The proposed method is evaluated on benchmark datasets and demonstrated to significantly improve performances. We introduce 3 new multiple reference SR datasets which will be made available to benchmark future SR approaches.

2. Related work

2.1. Single-image super-resolution (SISR)

A breakthrough in the SISR task was achieved when Dong *et al*. [9] tackled the problem with a convolutional neural network (CNN). From this work, the application of deep learning progressively replaced classic SR computer vision methods [37]. The pioneer work of Dong *et al*. [9] belongs to a group of SR methods that use mean squared error (MSE) as their objective function. VDSR [14] shows the importance of a deep layer architecture while SRResNet [15] and EDSR [19] demonstrate the benefit of using residual block [12] to alleviate the training. Several modifications of the residual structure such as skip connections [33], recursive structures [31] and channel-attention [43] further improved the accuracy of SISR. The state-of-the-art CSNLN [22] integrates a cross-scale non-local attention module to learn dependencies between the LR and HR images. Other works propose lightweight networks to alleviate computational cost [20, 23]. These residual networks ignore the human perception and only aim to high values of PSNR and SSIM, producing blurry SR images [37]. Generative adversarial networks (GANs), introduced in the SR task by Ledig *et al*. with SRGAN [15], aim to enhance the perceptual quality of the SR images. The performances of SRGAN were improved by ESRGAN [15], which replaces the adversarial loss with a relativistic adversarial loss. RSRGAN [42] develops a rank-content loss by training a ranker to obtain state-of-the-art visual results.

3D appearance super-resolution: There are only two deep learning works that super-resolve texture maps to enhance the appearance of 3D objects. The method proposed by Li *et al*. [18] processes, with a modified version of EDSR [19], LR texture maps and their normal maps to incorporate geometric information of the model in the learning. The pre-process to create normal maps introduces heavy computational cost. In the second work [25], a redundancy-based encoder generates a blurry texture map from LR images that is then deblurred by a SISR decoder. Its main objective is not the super-resolution but the creation of texture maps from a set of LR multi-view images.

2.2. Reference-based super-resolution (RefSR)

GANs were introduced to solve the problems of the residual networks by focusing on the perceptual quality of the image. However, their generative nature leads to the creation of unnatural textures in the SR image. RefSR approaches were applied to eliminate these artefacts by learning more accurate details from reference images. One of the first RefSR networks is CrossNet [46], which uses optical flow to align input and reference, limiting the matching of long distance correspondences. CrossNet was improved with two-stage cross-scale warping modules, adding to the optical flow alignment a further warping stage [32]. The optical flow introduces artefacts when misaligned references must be handled. The “patch-match” approach correlates the reference and input images by matching similar patches. In an early, non deep learning framework [6], patches of downsampled reference image are matched with gradient features of the LR image. This work was adapted by Zheng *et al*. [45] to perform semantic matching as well as to synthesise SR features through a CNN. More recently, Zhang *et al*. proposed SRNTT [44], which swaps the most similar features of the reference and the LR image through convolutional layers. TTSR [39] refines the matching mechanism by selecting and transferring only relevant textures from the reference image. SSEN [28] performs the patch-match through deformable convolution layers using an offset estimator. MASA [21] adds a spatial adaptation module to handle large disparity in color or luminance distribution between reference and input images. CIMR [8] is the only method in the literature that exploits multiple references. It selects a smaller subset from all the features of generic reference images without performing any comparison with the LR input, neglecting similar textures of the references. Our approach utilises a hierarchical patch-match method to search for relevant textures among all the feature vectors of multiple references, increasing the possibility to find more similar high-quality textures. It performs an attention-based similarity mapping between the references and subvectors of the LR input, improving performances. Finally, it significantly reduces the GPU usage of the patch-match approaches, facilitating the reproducibility of RefSR studies.

3. Adaptive multi-reference super-resolution

In this section we present the proposed AMRSR network, designed to exploit multiple HR reference images for training and inference whilst maintaining memory efficiency. A hierarchical attention-based approach is introduced for image feature matching from the LR input to the HR refer-

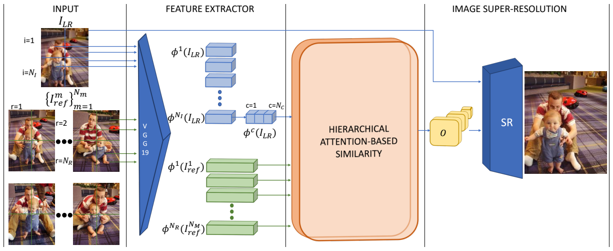

Figure 2: AMRSR inference overview. Given a LR input image and N_M reference images, AMRSR comprises three modules: feature extraction; hierarchical attention-based similarity; and image SR sampling. The resulting output is a HR reconstruction of the LR input image.

ence images using a perceptual loss. Hierarchical attention allows multiple HR reference images without a significant increase in GPU memory requirements and is demonstrated to improve performance versus a global similarity search (section 5.2). The problem of multiple-reference super-resolution can be stated as follows: given a LR input I_{LR} and a set of HR reference images $\{I_{ref}^m\}_{m=1}^{N_M}$, estimate a spatially coherent SR output I_{SR} with the structure of I_{LR} and the appearance detail resolution of the multiple-reference images. The SR output should contain perceptually plausible HR appearance detail without the introduction of visual artefacts such as blur, ringing or unnatural discontinuities observed with previous SR approaches for large up-scaling factors ($> 2\times$).

3.1. Overview of approach

Figure 2 presents an overview of the proposed approach, which comprises the following stages.

Feature Extraction: to reduce GPU memory consumption with multiple reference images, the LR input I_{LR} and HR reference images $\{I_{ref}^m\}_{m=1}^{N_M}$ are divided into N_I and N_R sub-parts, respectively. Image features are extracted from these parts using a pre-trained VGG-19 network [29].

Hierarchical Attention-based Similarity: computes a mapping of features from the LR image to the most similar features of the HR reference images. The similarity s_k is inferred between the feature vector of the LR input and of every reference image. The multiple references are then sampled based on the most similar features. This process is executed following a hierarchical structure of $l = N_L$ levels with an attention-based similarity mapping of the input feature vector. The output O is a feature vector containing the most similar reference features to the input features.

Image Super-resolution: given the feature similarity mapping O, a convolutional network super-resolves the LR input I_{LR} to obtain the SR output I_{SR} which maintains the spatial coherence of the input with the HR appearance detail of the reference images.

In contrast to the patch-match adopted by previous RefSR approaches [17, 21, 28, 39, 44], AMRSR performs feature similarity matching between subvectors of the LR input and reference images to focus the learning attention on the input features. A feature vector contains N features of the input image, each of which is a matrix of values that represents a specific image feature. Instead of processing the whole matrix, we divide it into submatrices and perform the similarity mapping with these. This improves the learning of the most similar features in each submatrix giving improved performance (Table 6). We refer the reader to the supplementary material (Table 2) for an outline of the notation.

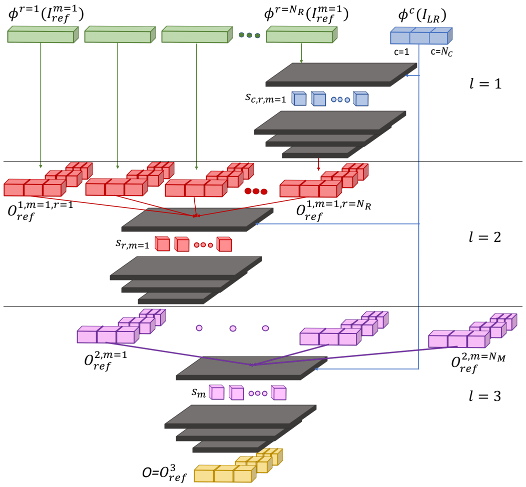

Figure 3: Hierarchical attention-based similarity with $N_L = 3$ levels.

3.2. Feature extraction

To conduct the similarity matching in the neural domain, feature vectors of the LR input and references must be retrieved. A problem of previous RefSR approaches is that the patch-match on HR reference images requires high GPU memory usage. To tackle this, we divide the input and reference images into N_I and N_R sub-parts, respectively. Feature vectors of these parts are extracted with a VGG-19 network [29] as shown in the second part of Figure 2. The feature vectors of each part of the input are divided into N_C subvectors $\{\{\phi_i^c(I_{LR})\}_{i=1}^{N_I}\}_{c=1}^{N_C}$ to perform an attention-based similarity on the LR input. For simplicity, we assume $N_I = 1$ (the input image is not divided into parts) and we express the set as $\phi^c(I_{LR})$ without loss of generality. If $N_I > 1$, the algorithm is repeated for each part and the outputs are concatenated. $N_M \times N_R$ feature vectors are retrieved for the references $\{\{\phi^r(I_{ref}^m)\}_{m=1}^{N_M}\}_{r=1}^{N_R}$ (expressed as $\phi^r(I_{ref}^m)$). Dividing the input and reference images into parts and inferring the similarity between them in a hierarchical order, establishes an efficient mechanism that improves performances and reduces GPU memory requirements. This is important for practical implementation of multi-reference SR within a fixed GPU memory size.

3.3. Hierarchical attention-based similarity

The objective of this stage is to map the features of the LR input to the most similar features of the HR reference images. The output is a feature vector that contains the values of these most similar reference features. A hierarchical approach of similarity mapping is performed over $l = N_L$ levels. For every level l of the hierarchy, a similarity map between LR input subvectors and reference features is computed. The most similar features are then retrieved considering the maximum values of the similarity map. A new feature vector is created with these features and used to compute the similarity map and the feature vector in the next level of the hierarchy.

The similarity map s_k^l for level l is evaluated by convolution between the subvectors $\phi^c(I_{LR})$ of the LR input and $O_{ref}^{l-1,r,m}$, which is either the input reference feature vectors $\phi^r(I_{ref}^m)$ if $l = 1$ or new vectors created in the level $l = l - 1$ (which contain features of the references $\{I_{ref}^m\}$):

$$s_k^l = \phi^c(I_{LR}) * \frac{P_k(O_{ref}^{l-1,r,m})}{||P_k(O_{ref}^{l-1,r,m})||} \tag{1}$$

$k = c$ if $l = 1$, $k = r$ or $k = m$ otherwise. P is the patch derived from the application of the patch-match approach: patches of $O_{ref}^{l-1,r,m}$ are convoluted with $\phi^c(I_{LR})$ to compute the similarity.

When the similarity map s_k^l is evaluated, a vector O_{ref}^l containing the most similar features of O_{ref}^{l-1} is created by applying either one of two distinct approaches:

1. **Input attention mapping** ($l = 1$): in the first level a feature vector is created by maximising over every subvector of the input:

$$O_{ref}^{1,r,m}(x,y) = P_{k^*}(\phi^r(I_{ref}^m))(x,y) \tag{2}$$
$$k^* = \underset{k=c}{\operatorname{argmax}}\, s_k^1(x,y)$$

   $O_{ref}^{1,r,m}(x,y)$ represents a single value in the (x,y) position of the created feature vector $O_{ref}^{1,r,m}$. This value corresponds to the (x,y) value of the k^* patch $P(\phi^r(I_{ref}^m))$ whose s^1 is the highest among all the similarity values $s_k^1(x,y)$ for each subvector of the LR input feature vector.

2. **Reference attention mapping** ($l > 1$): for subsequent levels of the hierarchy, a feature vector is created by maximising a new similarity s_k^l map over the feature vector created in the previous level.

$$O_{ref}^{l,k}(x,y) = O_{ref}^{l-1,k^*}(x,y) \tag{3}$$
$$k^* = \underset{k}{\operatorname{argmax}}\, s_k^l(x,y)$$

   $k = r$ or $k = m$ depending on which level is processed. The value of $O_{ref}^{l,k}$ in the (x,y) position is the value of $O_{ref}^{l-1,k}$ with the highest s^l among all the $s_k^l(x,y)$ of $O_{ref}^{l-1,k}$.

Mapping is repeated at multi-scales with three feature extractor levels to achieve robustness to the variance of colour and illumination [44]. The final output, obtained when the similarity mapping is performed for all the levels of the hierarchy, is a feature vector $O = O_{ref}^{N_L}$ which contains the features of the reference images that are most similar to every feature of the LR input. When the final level of the hierarchy is processed, N_K sets of weights W_k are computed as the maximum of the scalar product between $\phi^c(I_{LR})$ and the $O_{ref}^{l-1,k}$ vector produced in the previous level.

$$W_k = max(\phi^c(I_{LR}) \cdot O_{ref}^{l-1,k}) \tag{4}$$

The final set of weights W is then retrieved from these sets: the weight in position (x,y) of W has the same value of the weight in position (x,y) of the k^*-th set W_{k^*} with k^* from Equation 3 with $l = N_L$: $W(x,y) = W_{k^*}(x,y)$.

For $N_L = 3$ levels of hierarchy as shown in Figure 3:

$\mathbf{l} = \mathbf{1}$: feature similarity mapping between every subvector of the input vector and every part of every reference. *Input:* $\phi^c(I_{LR})$, $\phi^r(I_{ref}^m)$, $k = c$. *Output:* $\{\{O_{ref}^{1,m,r}\}_{m=1}^{N_M}\}_{r=1}^{N_R}$.

$\mathbf{l} = \mathbf{2}$: feature similarity mapping between the input subvectors and all the N_R parts of a single reference, repeated for every reference. *Input:* $\phi^c(I_{LR})$, $\{\{O_{ref}^{1,m,r}\}_{m=1}^{N_M}\}_{r=1}^{N_R}$, $k = r$. *Output:* $\{O_{ref}^{2,m}\}_{m=1}^{N_M}$.

$\mathbf{l} = \mathbf{3}$: feature similarity mapping between the input subvectors and all the references. *Input:* $\phi^c(I_{LR})$, $\{O_{ref}^{2,m}\}_{m=1}^{N_M}$, $k = m$. *Output*: $O = O_{ref}^3$.

3.4. Image super-resolution

In the last stage, I_{LR} is super-resolved with a generative network that exploits the information of the vectors obtained with the hierarchical similarity mapping whilst maintaining the spatial coherence of I_{LR}. These vectors are embedded to the input feature vector through channel-wise concatenation in different layers h of the network. We modified the architecture of the generator used by Zhang *et al.* [44] by eliminating the batch normalization layers since they can reduce the accuracy for dense pixel value predictions [40]. More details are explained in the supplementary material. A texture loss is defined to enforce the effect of the texture swapping between I_{LR} and the obtained O:

$$L_{tex} = \sum_h ||Gr(\phi_h(I_{SR} \cdot W_h)) - Gr(O_h \cdot W_h))|| \tag{5}$$

where $Gr(\cdot)$ computes the Gram matrix. Differently from [44], the weighting map W_h is computed among the N_M references. The weight of HR image features more similar to I_{LR} will be higher. Thus the appearance transfer from $\{I_{ref}\}$ to I_{SR} is adaptively enforced based on the references similarity. In addition, the network minimises:

- The adversarial loss L_{adv} to enhance the visual quality of the SR output. To stabilize the training, we use the WGAN-GP [11] for its gradient penalization feature.

- The reconstruction L_1 loss, since it has been demonstrated to give sharper performance than L_2 loss [39].
- The perceptual loss [13] L_p to enhance the similarity between the prediction and the target in feature space.

4. Dataset

To the best of our knowledge, no multi-reference benchmark datasets are available (only with a single reference [44]). To achieve our objective of multi-reference SR, we introduce three datasets:

1. CU4REF: this dataset is built from the single reference dataset CUFED5 [44]. 4 groups of images are defined from the CUFED dataset [36], each with a different similarity level from the LR input images. We use the images in these groups as our references. The training set contains 3957 groups of LR and reference images while the testing set contains 126 groups (4 references for every LR image).
2. HUMAP: to create the references of 67 synthetic human texture maps downloaded from several websites [2, 3, 4, 5], we import their 3D models in Blender [1] and render 8 camera views as reference images for each subject. Two real human texture maps retrieved from 16 multi-view images are added. Due to the low amount of data, we augment the dataset by cropping the texture maps into patches (256x256 size) [18]. The training dataset consists of 5505 groups of patches and references. The testing dataset comprises 336 groups created from 5 texture maps of 6 subjects (3 captured by 16 video-cameras, 2 using a 5x5 FLIR Grasshooper3 camera array) and a texture map of 2 people [7].
3. GEMAP: consists of generic LR texture maps associated with 8 references. The texture maps are taken from the 3DASR dataset [18], created from the multi-view images and 3D point clouds from other datasets ([16], [27], [26], [10], [47], [48]). The reference images for the texture maps of [16] are created with the same approach applied for HUMAP. For the other texture maps, the HR multi-view images captured by DSLR cameras are taken as references. The LR texture maps are cropped as in HUMAP. The training dataset contains 2032 groups and for testing 290 groups.

To evaluate the generalization capability of AMRSR on RGB images, we test it on Sun80 dataset [30], which has 80 natural images accompanied by a series of web-search references that significantly differ from the input images.

5. Results and evaluation

This section illustrates how AMRSR outperforms other state-of-the-art methods with quantitative and qualitative comparisons. Two ablation studies on the network configuration and on the advantage of multiple references are then presented. The GPU memory requirement of the state-of-the-art RefSR approaches is compared confirming the efficiency of AMRSR. The LR inputs are obtained by bicubic downscaling (4×) from their ground-truth HR images and the SR results are evaluated on PSNR and SSIM on the Y channel of YCbCr space. AMRSR parameters are: $N_M = 4$, $N_I = 1$, $N_R = 1$ for CU4REF dataset, $N_R = 16$ for all others (see Section 5.2). To integrate the input subvectors with the architecture structure, the value of N_C is:

$$N_C(x) = \frac{length(\phi(I_{LR})(relu3\_1))}{length(\phi(I_{LR})(reluq\_1)_{q=1}^{3})/4} \tag{6}$$

where q indicates the three different layers used for the multi-scale fashion approach. Further evaluations and results are presented in the supplementary material.

	Methods	CU4REF	Sun80	GEMAP	HUMAP
PSNR-oriented	SRResNet [15]	26.28/.7823	29.80/.8121	35.71/.9093	46.06/.9785
	RRDBNet [35]	26.22/.7828	29.56/.8053	35.77/.9102	46.25/.9790
	EDSR [19]	25.52/.7652	28.74/.7876	35.36/.9051	45.92/.9784
	MDSR [19]	26.43/.7822	29.96/.8137	35.84/.9107	46.06/.9784
	NHR [18]	–	–	33.13/.8981	36.15/.9544
	NLR [18]	–	–	33.13/.8941	42.22/.9731
	RCAN [43]	26.63/.7880	30.07/.8156	36.00/.9123	46.33/.9791
	MAFFSRN [23]	26.57/.7853	30.05/.8141	35.78/.9101	46.06/.9785
	CSNLN [22]	26.94/.7958	30.25/.8197	34.24/.9042	46.11/.9792
	SRNTT-l_2	27.62/.8201	30.16/.8176	35.91/.9120*	46.28/.9791*
	TTSR-l_2	27.02/.8001	29.97/.8123	35.35/.9083*	37.50/.9709*
	MASA-l_2	27.49/.8145	30.42/.8263	35.53/.9046*	46.11/.9784*
	AMRSR-l_2	**28.32/.8394**	**30.95/.8438**	**36.82/.9248**	**46.86/.9814**
Visual-Oriented	SRGAN [15]	23.63/.6761	25.97/.6570	31.56/.8551	41.68/.9525
	ESRGAN [35]	23.69/.6884	26.42/.7005	32.34/.8664	42.78/.9669
	RSRGAN [42]	25.49/.7494	29.10/.7873	33.90/.8892	43.44/.9707
	CrossNet [46]	26.00/.7576	29.16/.7834	32.95/.8741	30.30/.9317
	SSEN [28]	22.71/.7169	26.58/.7824	25.86/.8150	35.61/.9446
	SRNTT [44]	26.42/.7738	29.72/.7984	34.78/.8963*	45.03/.9743*
	TTSR [39]	25.59/.7645	28.23/.7595	34.22/.8912*	35.32/.9566*
	MASA [21]	24.84/.7311	27.16/.7129	34.38/.8850*	43.27/.9598*
	AMRSR	27.49/.8145	30.42/.8263	35.80/.9122	45.56/.9771

Table 1: PSNR/SSIM values of different SR approaches. * indicates that the references are downscaled of a factor of 2 (see Section5.2).

5.1. Comparison with state-of-the-art approaches

Qualitative and quantitative comparisons are performed with state-of-the-art SISR and RefSR approaches. The SISR methods are the PSNR-oriented EDSR [19], MDSR [19], RRDBNet [35], SRResNet [15], RCAN [43], NHR [18], NLR [18], CSNLN [22], MAFFSRN [23] and the visual-oriented SRGAN [15], ESRGAN [35], RSRGAN [42]. The RefSR approaches are CrossNet [46], SSEN [28], SRNTT [44], TTSR [39] and MASA [21] (published June '21). We train each network with the datasets presented in Section 4 with the same training configurations. Training with adversarial loss usually deteriorates the quantitative results. For a fair comparison with the PSNR-oriented methods, we train our model, SRNTT, MASA and TTSR only on reconstruction loss (named with the suffix "l_2"). NHR and NLR are tested on HUMAP and GEMAP since they require normal maps (retrieved with Blender [1]).

Quantitative comparison: the PSNR and SSIM values of each method are presented in Table 1, which is divided into two parts: PSNR-oriented networks in the upper part; visual-oriented GANs and RefSR in the lower part. The highest scores are highlighted in red while the second highest scores are blue. The bold red figures are the highest across both PSNR- and visual-oriented methods. AMRSR-

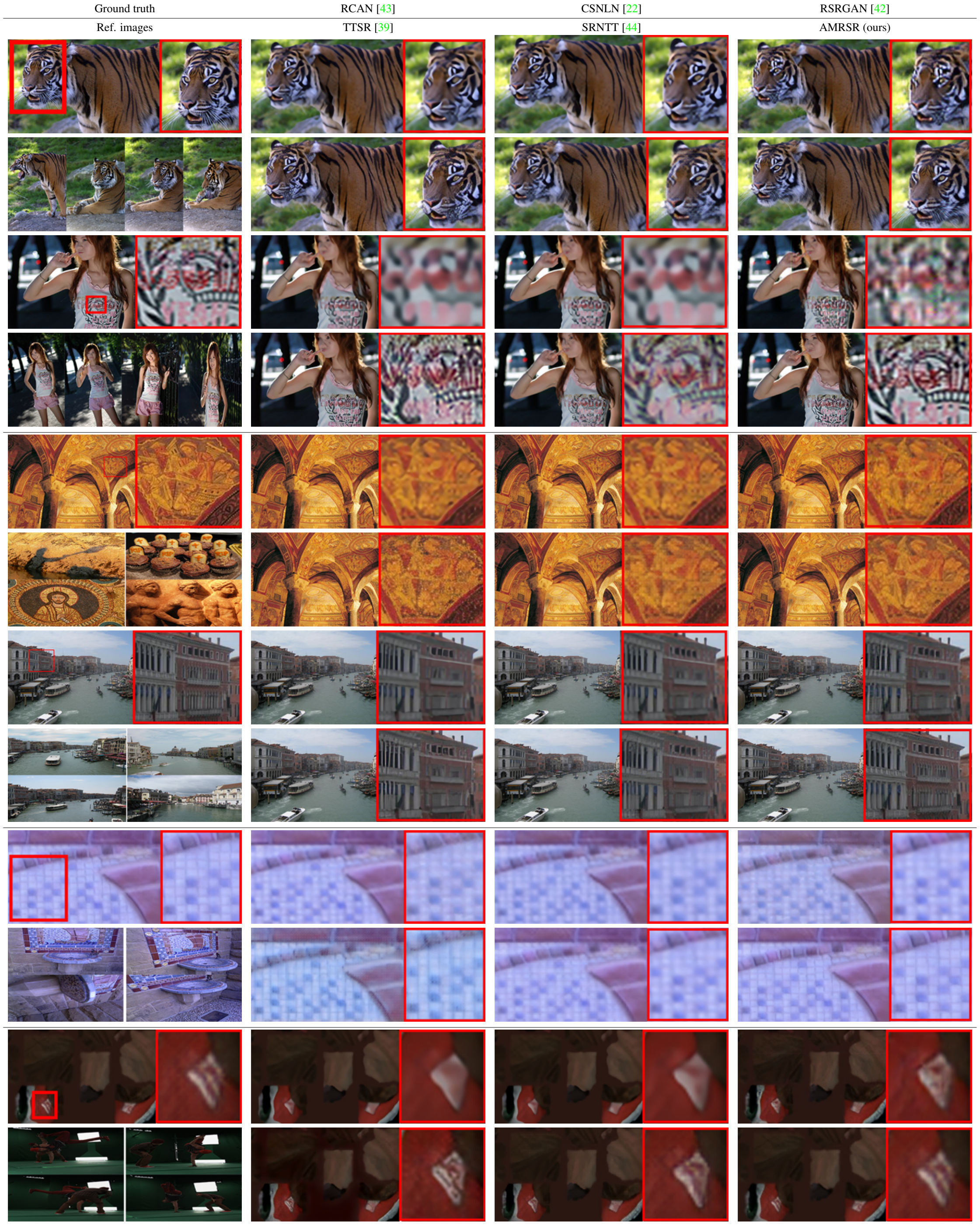

Figure 4: Qualitative comparison among the state-of-the-art SR approaches on CU4REF (first two examples), Sun80 (third and fourth), GEMAP (fifth) and HUMAP (last). The references are shown for each example. The top left or the most left reference was used in the single-reference SR approaches.

l_2 achieves the highest values of PSNR and SSIM in the PSNR- and visual-oriented methods for all four datasets.

Qualitative comparison: Figure 4 shows SR examples of the most relevant approaches considered in our evaluation. The SR outputs produced by the PSNR-oriented methods (RCAN, CSNLN) are blurrier and the details are less sharp. The results produced by the visual-oriented approaches (RSRGAN, SRNTT, TTSR) present unpleasant artefacts such as ringing and unnatural discontinuities. The SR outputs of AMRSR are less blurry with sharper and finer details as shown in the zoomed patches of the examples, whose quality is higher than the other results.

User study evaluation: to further evaluate the visual quality of the SR outputs, we conduct a user study, where AMRSR is compared with five approaches. 50 random SR output pairs were shown to 60 subjects. Each pair consists

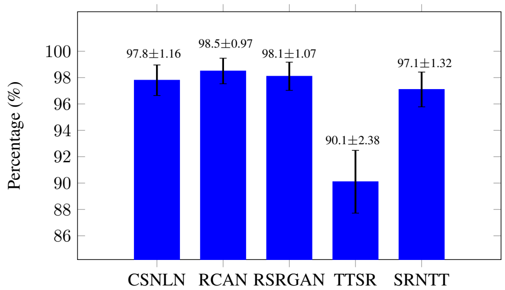

Figure 5: Percentage of users that prefer AMRSR over 5 approaches in the user study. The error bars indicate the 95% confidence interval.

Algorithms	**L1**	**L2**	**L3**	**L4**
Cross-Net [46]	25.98/.7582	25.98/.7582	25.97/.7581	25.97/.7581
SSEN [28]	22.71/.7169	22.43/.7114	22.30/.7131	22.13/.7084
SRNTT [44]	26.42/.7738	26.34/.7690	26.27/.7682	26.24/.7678
TTSR [39]	25.59/.7645	25.08/.7442	24.98/.7414	24.95/.7412
MASA [21]	24.84/.7311	24.27/.7093	24.25/.7077	24.23/.7057
AMRSR 1ref	26.77/.7882	26.71/.7869	26.63/.7841	26.48/.7804
SRNTT-l_2	27.62/.8201	27.21/.8039	27.05/.8003	26.92/.7969
TTSR-l_2	27.02/.8001	26.48/.7809	26.40/.7792	26.35/.7784
MASA-l_2	27.49/.8145	26.66/.7881	26.60/.7863	26.55/.7843
AMRSR 1-l_2	**27.92/.8293**	**27.51/.8152**	**27.40/.8114**	**27.24/.8080**

Table 2: Quantitative comparison between AMRSR and RefSR methods using single reference with different levels of similarity to the LR input.

of an image of AMRSR and the counterpart generated by one of the other approaches. The users were asked to pick the image with the best visual quality. The values on the Y-axis of Figure 5 illustrate the percentage of the users that select AMRSR outputs. AMRSR significantly outperforms the other methods with over 90% of users voting for it.

Influence of dissimilar references: similarity between LR and reference images significantly influences the performances of RefSR methods [44]. Following the setting in [44], we evaluate the effect of dissimilar reference images to the LR input testing on four similarity levels (from L1 to L4) of the CUFED5 test set, defined by computing the number of best matches of SIFT features between the input image and the references. The references that belong to L1 have the highest number of matches while the L4 ones have the lowest number. The highest figures of PSNR and SSIM (Table 2) are obtained by our network when a single reference is used even though its level of similarity has decreased, confirming the higher efficiency of AMRSR when the references are not similar to the LR input. To show the efficiency of leveraging multiple references, we evaluate the RefSR approaches on the CU4REF dataset by swapping the references of an image with other images in the dataset. The quantitative results of Table 6 and the visual comparison in Figure 3 demonstrate the benefits of using multiple dissimilar references. AMRSR is able to find more similar patches within multiple references even if these are very dissimilar to the LR input. For dissimilar references, AMRSR achieves higher PSNR and SSIM than other methods and than when a single reference is used (AMRSR 1). This study demonstrates the performance of adaptive sampling of AMRSR even for references with dissimilar features.

Comparison with CIMR [8]: we compare AMRSR with CIMR [8], the only other multi-reference super-resolution approach. AMRSR is trained following the setting in [8] on CUFED5 dataset (13,761 images). We randomly associated N_M references taken from Outdoor Scene (OST) dataset [34] to each LR input image. CIMR is evaluated on content-independent references using a reference pool to select a subset of feature vectors from 300 reference images. We evaluate AMRSR by randomly associating to the LR input images N_M reference images from the 300 images. Results presented in Table 4 show that AMRSR outperforms CIMR in the multiple references case.

	TTSR [39]	SRNTT [44]	CrossNet [46]	SSEN [28]	MASA [21]	AMRSR 1	AMRSR
Visual	25.08/.746	26.17/.765	25.98/.758	22.83/.7148	24.17/.703	26.36/.775	**26.47/.780**
PSNR	26.59/.784	26.67/.790	–	–	26.42/.780	27.06/.805	**27.10/.806**

Table 3: PSNR/SSIM values with references dissimilar to the LR input.

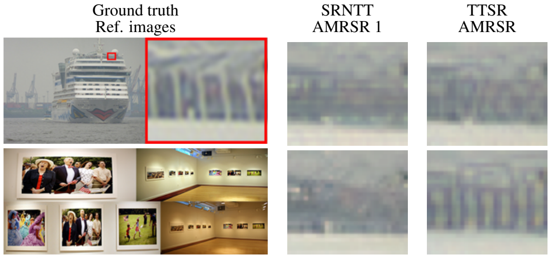

Figure 6: Visual comparison of SR outputs obtained with multiple references dissimilar to the LR input.

Algorithms	**Visual-oriented**		**PSNR-oriented**	
	CUFED5	Sun80	CUFED5	Sun80
SRNTT [44]	25.61/.764	27.59/.756	26.24/.784	28.54/.793
CIMR [8]	26.16/.781	29.67/.806	26.35/.789	30.07/.813
AMRSR $N_M = 4$	26.87/.795	30.53/.829	27.57/.820	31.10/.847
AMRSR $N_M = 8$	26.92/.796	30.64/.832	27.63/.821	31.24/.851
AMRSR $N_M = 16$	26.98/.797	30.69/.834	27.69/.823	31.29/.852
AMRSR $N_M = 32$	27.01/.798	30.75/.835	27.75/.824	31.35/.854
AMRSR $N_M = 64$	**27.08/.800**	**30.80/.836**	**27.81/.826**	**31.41/.854**

Table 4: Quantitative comparison between AMRSR and CIMR exploiting multiple reference images. The results of CIMR are taken from [8].

5.2. Ablation studies

Number of reference images: a key contribution of our work is the transfer of high-quality textures from multiple references to increase the matching between similar LR input and HR reference features. To prove this, we test AMRSR by changing the number of references. The results of using $N_M = 1, 2, 4$ reference images are compared for CU4REF and Sun80 datasets. $N_M = 8$ references are also considered for HUMAP and GEMAP. Table 5 presents the PSNR and SSIM figures for the different cases, including the second best results of related works ("2^{nd} best"). Increasing the number of references leads to higher values of PSNR and SSIM. The highest ones are generally obtained with the maximum number of references. AMRSR outperforms the 2^{nd} best techniques also when a single reference is used. Figure 8 confirms the advantage of using multiple references. The hairs of the human subject in the texture map example are sharper when 4 references are used because they are transferred from the side of the model, which is not visible with 1 or 2 references. Similarly, the window

	Nr. references	CU4REF	Sun80	GEMAP	HUMAP
PSNR-Oriented	1 Reference	27.92/.8293	30.61/.8376	36.49/.9219	46.64/.9803
	2 References	28.26/.8384	30.79/.8380	36.58/.9230	46.74/.9808
	4 References	**28.32/.8394**	**30.95/.8438**	36.82/.9248	46.86/.9814
	8 References	–	–	**36.84/.9255**	**46.87/.9814**
	2^{nd} best	27.62/.8201	30.25/.8197	36.00/.9123	46.33/.9792

Table 5: PSNR/SSIM values obtained with different numbers of reference images. The bottom row shows the figures for the second-best approaches.

	Config.	CU4REF	Sun80	GEMAP	HUMAP
PSNR-Oriented	No attention	27.54/.8170	30.18/.8171	35.91/.9130	46.33/.9791
	Ref. attention	27.55/.8179	30.21/.8196	35.93/.9131	46.43/.9794
	Both attention	27.18/.8064	30.24/.8194	35.91/.9130	46.43/.9793
	AMRSR	**28.32/.8394**	**30.95/.8438**	**36.82/.9248**	**46.86/.9814**

Table 6: PSNR/SSIM values of different configurations of AMRSR obtained by dividing into subvectors the feature vectors of references (ref), of both reference and input (both) or none (no).

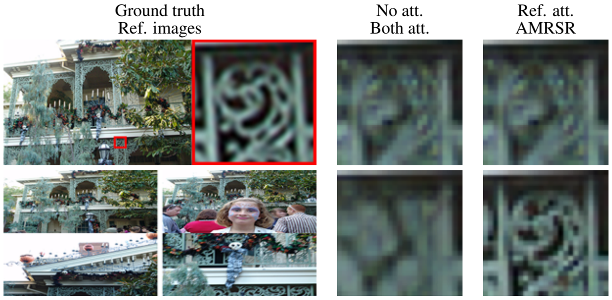

Figure 7: Visual comparison of the results obtained by changing attention.

grates are neat only when 4 references are used because the network learns their texture from the third reference.

Attention mapping: we evaluate the effect of processing subvectors of the LR input in the similarity mapping by comparing the actual configuration of AMRSR with three others: (i) without any attention mapping (processing the whole input vector); (ii) with an attention mapping in the reference feature vectors (dividing them into subvectors); (iii) with an attention mapping in both the input and reference feature vectors. The obtained values of PSNR and SSIM are shown in Table 6. The effectiveness of the input attention mapping is proved by both the quantitative and qualitative evaluation, with a significant boost in the performance and higher quality of the SR examples of Figure 7.

Part-based mechanism and GPU memory usage: we finally evaluate the effect of the part-based mechanism of AMRSR by computing the required GPU memory of a Quadro RTX 8000 GPU (48 GB) during inference and comparing it with TTSR, SRNTT, MASA, AMRSR with different numbers of references and with different values of N_R (Table 7). Changing the value of N_R leads to a modification of the hierarchical structure: if $N_R = 1$, the 2^{nd} level of the hierarchy is skipped. The maximum size of the reference images is: (500x500) for CU4REF, (1024x928) for Sun80, (5184x3840) for HUMAP and (4032x6048) for GEMAP. Increase in the reference image size results in increased memory consumption and improve super-resolution quality. AMRSR requires significantly less memory than the single RefSR approaches when multiple references are used. To test TTSR, SRNTT and MASA with HUMAP and GEMAP datasets, the reference must be downscaled ($2\times$) to consume less than 48GB. For a fair comparison, we test our network with 1 and 4 downscaled references ("cut" in the table). The comparison between different choices of N_R shows that, when higher resolution references are exploited with higher values of N_R, the memory footprint is reduced and the PSNR and SSIM are increased. For CU4REF, the best performances are achieved with $N_R = 1$ because the size of its references is much lower.

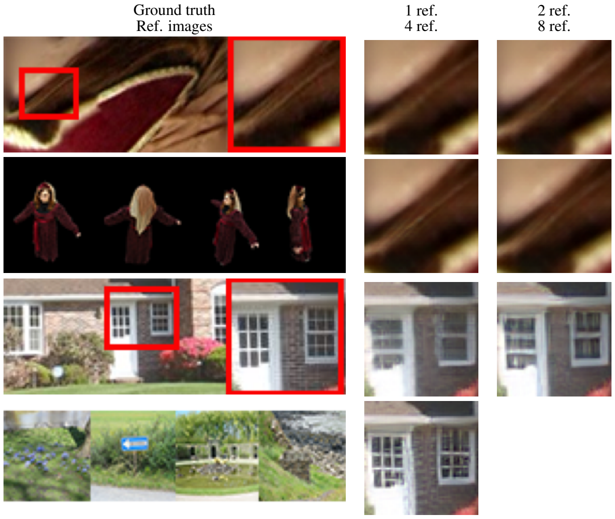

Figure 8: Visual comparison of the results obtained by changing the number of references. The first example is of HUMAP, the other is of Sun80.

Algorithms	CU4REF		Sun80		GEMAP		HUMAP	
	PSNR/SSIM	GPU	PSNR/SSIM	GPU	PSNR/SSIM	GPU	PSNR/SSIM	GPU
AMRSR 1	27.92/.829	**1.01**	30.61/.837	**2.10**	36.49/.921	15.57	46.64/.980	11.47
AMRSR 2	28.26/.838	1.25	30.79/.838	2.66	36.56/.922	15.63	46.74/.980	11.53
AMRSR 8	–	–	–	–	**36.84/.925**	15.80	**46.87/.981**	11.70
$N_R = 1$	**28.32/.839**	1.36	30.84/.839	3.96	36.60/.922	40.15	46.81/.981	29.60
$N_R = 4$	28.00/.835	1.21	30.87/.841	3.23	36.65/.923	28.49	46.85/.981	20.98
$N_R = 16$	27.97/.834	1.22	**30.95/.843**	3.24	36.82/.924	15.69	46.86/.981	11.59
cut 1	–	–	–	–	36.44/.920	**3.99**	46.63/.980	**2.94**
cut 4	–	–	–	–	36.75/.924	4.10	46.81/.981	3.06
SRNTT [44]	27.62/.820	2.81	30.16/.817	14.63	35.91/.912	(26.51)	46.28/.979	(19.49)
TTSR [39]	27.02/.800	4.24	29.97/.812	20.61	35.35/.908	(40.07)	37.50/.970	(29.47)
MASA [21]	27.49/.814	8.23	30.42/.826	15.69	35.53/.904	(42.11)	46.11/.978	(31.49)

Table 7: PSNR/SSIM values and GPU memory usage (in GB) of different configurations of AMRSR and the state-of-the-art RefSR approaches. N_R is the number of parts which the reference images are divided into.

6. Conclusion

In this paper, we tackle the super-resolution problem with a multiple-reference super-resolution network that is able to transfer more plausible textures from several references to the super-resolution output. Our network focuses the learning attention in the comparison between subvectors of the low-resolution input and the reference feature vectors, achieving significant qualitative and quantitative improvements as demonstrated from the evaluation. A hierarchical part-based mechanism is introduced to reduce the GPU memory usage, which is prohibitive if previous RefSR methods are applied with high-resolution reference images. In addition, we introduce 3 datasets to facilitate the research for multiple-reference and 3D appearance super-resolution.

Acknowledgement: This research was supported by UKRI EPSRC Platform Grant EP/P022529/1.

References

[1] Blender. https://www.blender.org/. Accessed: 2020-12-26. 5

[2] Cgtrader. hhttps://www.cgtrader.com/free-3d-models. Accessed: 2020-12-26. 5

[3] Free3d. https://free3d.com/3d-models/. Accessed: 2020-12-26. 5

[4] Renderpeople. https://renderpeople.com/. Accessed: 2020-12-26. 5

[5] Turbosquid. https://www.turbosquid.com/Search/3D-Models/free. Accessed: 2020-12-26. 5

[6] Vivek Boominathan, Kaushik Mitra, and Ashok Veeraraghavan. Improving resolution and depth-of-field of light field cameras using a hybrid imaging system. In *2014 IEEE International Conference on Computational Photography (ICCP)*, pages 1–10. IEEE, 2014. 1, 2

[7] Alvaro Collet, Ming Chuang, Pat Sweeney, Don Gillett, Dennis Evseev, David Calabrese, Hugues Hoppe, Adam Kirk, and Steve Sullivan. High-quality streamable free-viewpoint video. *ACM Transactions on Graphics (ToG)*, 34(4):1–13, 2015. 5

[8] Shuguang Cui. Towards content-independent multi-reference super-resolution: Adaptive pattern matching and feature aggregation. 2020. 2, 7

[9] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *European conference on computer vision*, pages 184–199. Springer, 2014. 2

[10] Bastian Goldlücke, Mathieu Aubry, Kalin Kolev, and Daniel Cremers. A super-resolution framework for high-accuracy multiview reconstruction. *International journal of computer vision*, 106(2):172–191, 2014. 5

[11] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in neural information processing systems*, pages 5767–5777, 2017. 4

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2

[13] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016. 5

[14] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1646–1654, 2016. 2

[15] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017. 2, 5

[16] Andreas Ley, Ronny Hänsch, and Olaf Hellwich. Syb3r: A realistic synthetic benchmark for 3d reconstruction from images. In *European Conference on Computer Vision*, pages 236–251. Springer, 2016. 5

[17] Kai Li, Shenghao Yang, Runting Dong, Xiaoying Wang, and Jianqiang Huang. Survey of single image super-resolution reconstruction. *IET Image Processing*, 14(11):2273–2290, 2020. 3

[18] Yawei Li, Vagia Tsiminaki, Radu Timofte, Marc Pollefeys, and Luc Van Gool. 3d appearance super-resolution with deep learning. In *In Proceedings of the IEEE International Conference on Computer Vision*, 2019. 2, 5

[19] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 136–144, 2017. 2, 5

[20] Jie Liu, Jie Tang, and Gangshan Wu. Residual feature distillation network for lightweight image super-resolution. *arXiv preprint arXiv:2009.11551*, 2020. 2

[21] Liying Lu, Wenbo Li, Xin Tao, Jiangbo Lu, and Jiaya Jia. Masa-sr: Matching acceleration and spatial adaptation for reference-based image super-resolution. *arXiv preprint arXiv:2106.02299*, 2021. 2, 3, 5, 7, 8

[22] Yiqun Mei, Yuchen Fan, Yuqian Zhou, Lichao Huang, Thomas S Huang, and Honghui Shi. Image super-resolution with cross-scale non-local attention and exhaustive self-exemplars mining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5690–5699, 2020. 1, 2, 5, 6

[23] Abdul Muqeet, Jiwon Hwang, Subin Yang, Jung Heum Kang, Yongwoo Kim, and Sung-Ho Bae. Ultra lightweight image super-resolution with multi-attention layers. *arXiv preprint arXiv:2008.12912*, 2020. 2, 5

[24] Garima Pandey and Umesh Ghanekar. A compendious study of super-resolution techniques by single image. *Optik*, 166:147–160, 2018. 1

[25] Audrey Richard, Ian Cherabier, Martin R Oswald, Vagia Tsiminaki, Marc Pollefeys, and Konrad Schindler. Learned multi-view texture super-resolution. In *2019 International Conference on 3D Vision (3DV)*, pages 533–543. IEEE, 2019. 2

[26] Thomas Schops, Johannes L Schonberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3260–3269, 2017. 5

[27] Steven M Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, volume 1, pages 519–528. IEEE, 2006. 5

[28] Gyumin Shim, Jinsun Park, and In So Kweon. Robust reference-based super-resolution with similarity-aware deformable convolution. In *Proceedings of the IEEE/CVF Con-*

ference on Computer Vision and Pattern Recognition, pages 8425–8434, 2020. 2, 3, 5, 7

[29] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 3

[30] Libin Sun and James Hays. Super-resolution from internet-scale scene matching. In *2012 IEEE International Conference on Computational Photography (ICCP)*, pages 1–12. IEEE, 2012. 5

[31] Ying Tai, Jian Yang, and Xiaoming Liu. Image super-resolution via deep recursive residual network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3147–3155, 2017. 2

[32] Yang Tan, Haitian Zheng, Yinheng Zhu, Xiaoyun Yuan, Xing Lin, David Brady, and Lu Fang. Crossnet++: Cross-scale large-parallax warping for reference-based super-resolution. *IEEE Computer Architecture Letters*, (01):1–1, 2020. 2

[33] Tong Tong, Gen Li, Xiejie Liu, and Qinquan Gao. Image super-resolution using dense skip connections. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4799–4807, 2017. 2

[34] Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 606–615, 2018. 7

[35] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018. 5

[36] Yufei Wang, Zhe Lin, Xiaohui Shen, Radomir Mech, Gavin Miller, and Garrison W Cottrell. Event-specific image importance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4810–4819, 2016. 5

[37] Zhihao Wang, Jian Chen, and Steven CH Hoi. Deep learning for image super-resolution: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 1, 2

[38] Chih-Yuan Yang, Chao Ma, and Ming-Hsuan Yang. Single-image super-resolution: A benchmark. In *European Conference on Computer Vision*, pages 372–386. Springer, 2014. 1

[39] Fuzhi Yang, Huan Yang, Jianlong Fu, Hongtao Lu, and Baining Guo. Learning texture transformer network for image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5791–5800, 2020. 2, 3, 5, 6, 7, 8

[40] Jiahui Yu, Yuchen Fan, Jianchao Yang, Ning Xu, Zhaowen Wang, Xinchao Wang, and Thomas Huang. Wide activation for efficient and accurate image super-resolution. *arXiv preprint arXiv:1808.08718*, 2018. 4

[41] Huanjing Yue, Xiaoyan Sun, Jingyu Yang, and Feng Wu. Landmark image super-resolution by retrieving web images. *IEEE Transactions on Image Processing*, 22(12):4865–4878, 2013. 1

[42] Wenlong Zhang, Yihao Liu, Chao Dong, and Yu Qiao. Ranksrgan: Generative adversarial networks with ranker for image super-resolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3096–3105, 2019. 1, 2, 5, 6

[43] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 286–301, 2018. 2, 5, 6

[44] Zhifei Zhang, Zhaowen Wang, Zhe Lin, and Hairong Qi. Image super-resolution by neural texture transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7982–7991, 2019. 1, 2, 3, 4, 5, 6, 7, 8

[45] Haitian Zheng, Mengqi Ji, Lei Han, Ziwei Xu, Haoqian Wang, Yebin Liu, and Lu Fang. Learning cross-scale correspondence and patch-based synthesis for reference-based super-resolution. In *BMVC*, 2017. 2

[46] Haitian Zheng, Mengqi Ji, Haoqian Wang, Yebin Liu, and Lu Fang. Crossnet: An end-to-end reference-based super resolution network using cross-scale warping. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 88–104, 2018. 1, 2, 5, 7

[47] Qian-Yi Zhou and Vladlen Koltun. Color map optimization for 3d reconstruction with consumer depth cameras. *ACM Transactions on Graphics (TOG)*, 33(4):1–10, 2014. 5

[48] Michael Zollhöfer, Angela Dai, Matthias Innmann, Chenglei Wu, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Shading-based refinement on volumetric signed distance functions. *ACM Transactions on Graphics (TOG)*, 34(4):1–14, 2015. 5