This ICCV workshop paper is the Open Access version, provided by the Computer Vision Foundation. Except for this watermark, it is identical to the accepted version; the final published version of the proceedings is available on IEEE Xplore.

An Experimental Protocol for Neural Architecture Search in Super-Resolution

Jesús Leopoldo Llano García Raúl Monroy Víctor Adrián Sosa Hernández
Tecnologico de Monterrey, School of Engineering and Sciences
Av. Lago de Guadalupe Km 3.5, Atizapán de Zaragoza, Edo. Mexico 52926, MEXICO
{A01748867, raulm, vsosa}@tec.mx

Abstract

Neural architecture search has seen continual progress due to the interest in automating architecture design in deep learning following the promise of finding the best possible neural network architecture tailored for a particular task. Recently, many works focused on tackling tasks like image classification and language modeling, allowing significant developments in computer vision and NLP. As research in such directions has established standard criteria and benchmarking tasks for algorithmic performance comparison, the same cannot be said of other applications and tasks. Our work presents an experimental comparison protocol that narrows down the process of evaluating super-resolution image restoration architectures in neural architecture search approaches. Such protocol consists of two datasets for training and validation during and after the architecture search, and the application of a Bayesian statistical test for studying the observable results.

1. Introduction

Deep learning rapidly became a predominant tool in many fields of machine learning due to its impressive performance on unstructured data, as different authors demonstrate that the structure of a network allows the extraction of dominant features from data [22]. The automatic feature extraction ingrained within deep models makes architectural design crucial to ensure good predictive performance, where extensive area-specific knowledge and experience are necessary for designing the proper neural architecture structures by hand. Even then, constructing a network becomes highly dependent on time-consuming trial-and-error processes, in which inherent limitations of human knowledge make it difficult for researchers to leave behind preconceived notions of well-studied and well-performing known paradigms used on specific tasks. The field of Neural Architecture Search (NAS) originates in an attempt to alleviate the cumbersome process of deep architecture design by automating the discovery of novel structures, ideally reducing any human intervention.

In the past, NAS has found broad success at discovering innovative models capable of surpassing hand-crafted networks, focusing mainly on computer vision and natural language processing tasks [24, 9, 10], yet experts claim that the true potential of NAS is yet to be studied in broader areas of application [24, 9]. The ever-increasing demand for intelligent image processing, striving to achieve better image representations with more detail even from poor capturing conditions, has motivated experts to study the possibility of addressing the problem of Super-Resolution Image Restoration (SRIR) from a NAS perspective. The SRIR task tries to predict a high-resolution image from one or more low-resolution samples, thus improving the visual information of images after capture [16]. As a central task within image restoration, SRIR has found relevance in various applications ranging from medical image processing [2, 15] to surveillance and security [18], being able to help other models reach higher performance by improving perceptual quality before and after training.

While it is indubitable the success that NAS approaches have, being an emerging field, it still poses many issues in need of a solution if we wish for research in this direction to maintain momentum. One such challenge is the process of machine-crafted model evaluation for pipeline comparison. In the current state of the art, it has become the norm that novel developments are only compared by contrasting any found architectures against others, either machine- or hand-crafted. A priory, direct empirical comparisons can help motivate continual improvements in both NAS and Deep Learning (DL) while pushing forward any existing baseline. Yet, in the longer term, this could prove insufficient to demonstrate the advantages that some NAS pipelines could have against competing approaches or even human-driven architectural design. Several contemporary works that study the application of NAS in the context of image restoration tasks follow very narrow experimental protocols, contrasting architectures just by evaluating final models over benchmarking tasks [17, 20]. The narrow scope presents more than enough reasons to motivate a necessity for incorporat-

ing additional elements as part of experimental protocols, such as standardized data sets for training and testing of machine-crafted models and statistical analyses of any obtained result as part of experimentation.

Inspired by recurring trends emerging within various scientific areas that rely on empirical observations, we introduce in this manuscript what we believe are the first steps towards achieving a fair experimental contrasting protocol for machine-crafted DL models improving the discussion of results for NAS applied to SRIR in Section 2. Our proposal delimits the tasks that will serve for both the training and testing of machine-crafted models as well as the metrics used for both direct and statistical contrast of the performance and efficiency of models. With this, we seek to motivate future experimental protocols to consider the application of more advanced inferential statistics while quantifying the significance of presented results, beyond merely discussing the observable differences found among competing approaches. Moreover, the simplicity of this protocol aims at extending the contemporary standards of NAS applied to SRIR, while maintaining enough simplicity to be further extended and applied in later instances.

The proposed comparison protocol is described piecewise in the following sections of this article, starting with the specifics of the data used for training and testing machine-crafted models in Section 3, the necessary information on the use and interpretation of a Bayesian signed-rank test for statistical analysis of results as presented in work [3] can be found in Section 4. This comparison exclusively deals with the performance and complexity of found architectures. Lastly, remarks on the importance of this kind of research and conclusions are located in Section 5.

All data sets and codes necessary for the application of this protocol are publicly available at `https://github.com/jesusllg/An-Experimental-Protocol-for-NAS-SR.git`.

Deep learning rapidly became a predominant tool in many fields of machine learning due to its impressive performance on unstructured data, as different authors demonstrate that the structure of a network allows the extraction of dominant features from data [22]. The automatic feature extraction ingrained within deep models makes architectural design crucial to ensure good predictive performance, where extensive area-specific knowledge and experience are necessary for designing the proper neural architecture structures by hand. Even then, constructing a network becomes highly dependent on time-consuming trial-and-error processes, in which inherent limitations of human knowledge make it difficult for researchers to leave behind preconceived notions of well-studied and well-performing known paradigms used on specific tasks. The field of Neural Architecture Search (NAS) originates in an attempt to alleviate the cumbersome process of deep architecture design by automating the discovery of novel structures, ideally reducing any human intervention.

In the past, NAS has found broad success at discovering innovative models capable of surpassing hand-crafted networks, focusing mainly on computer vision and natural language processing tasks [24, 9, 10], yet experts claim that the true potential of NAS is yet to be studied in broader areas of application [24, 9]. The ever-increasing demand for intelligent image processing, striving to achieve better image representations with more detail even from poor capturing conditions, has motivated experts to study the possibility of addressing the problem of Super-Resolution Image Restoration (SRIR) from a NAS perspective. The SRIR task tries to predict a high-resolution image from one or more low-resolution samples, thus improving the visual information of images after capture [16]. As a central task within image restoration, SRIR has found relevance in various applications ranging from medical image processing [2, 15] to surveillance and security [18], being able to help other models reach higher performance by improving perceptual quality before and after training.

While it is indubitable the success that NAS approaches have, being an emerging field, it still poses many issues in need of a solution if we wish for research in this direction to maintain momentum. One such challenge is the process of machine-crafted model evaluation for pipeline comparison. In the current state of the art, it has become the norm that novel developments are only compared by contrasting any found architectures against others, either machine- or hand-crafted. A priory, direct empirical comparisons can help motivate continual improvements in both NAS and Deep Learning (DL) while pushing forward any existing baseline. Yet, in the longer term, this could prove insufficient to demonstrate the advantages that some NAS pipelines could have against competing approaches or even human-driven architectural design. Several contemporary works that study the application of NAS in the context of image restoration tasks follow very narrow experimental protocols, contrasting architectures just by evaluating final models over benchmarking tasks [17, 20]. The narrow scope presents more than enough reasons to motivate a necessity for incorporating additional elements as part of experimental protocols, such as standardized data sets for training and testing of machine-crafted models and statistical analyses of any obtained result as part of experimentation.

Inspired by recurring trends emerging within various scientific areas that rely on empirical observations, we introduce in this manuscript what we believe are the first steps towards achieving a fair experimental contrasting protocol for machine-crafted DL models improving the discussion of results for NAS applied to SRIR in Section 2. Our proposal delimits the tasks that will serve for both the training

and testing of machine-crafted models as well as the metrics used for both direct and statistical contrast of the performance and efficiency of models. With this, we seek to motivate future experimental protocols to consider the application of more advanced inferential statistics while quantifying the significance of presented results, beyond merely discussing the observable differences found among competing approaches. Moreover, the simplicity of this protocol aims at extending the contemporary standards of NAS applied to SRIR, while maintaining enough simplicity to be further extended and applied in later instances.

The proposed comparison protocol is described piecewise in the following sections of this article, starting with the specifics of the data used for training and testing machine-crafted models in Section 3, the necessary information on the use and interpretation of a Bayesian signed-rank test for statistical analysis of results as presented in work [3] can be found in Section 4. This comparison exclusively deals with the performance and complexity of found architectures. Lastly, remarks on the importance of this kind of research and conclusions are located in Section 5.

All data sets and codes necessary for the application of this protocol are publicly available and will be included as part of this manuscript later to preserve anonymity during the Double-blind peer review process.

2. Protocol for evaluating SRIR NAS methods

Evaluating machine- or hand-crafted model performance requires various elements to ensure an adequate and fair comparison among different approaches. The complexity of result replicability and the expense of empirical analysis of NAS pipelines has forced many current works, especially in SRIR, to compare results by direct contrast of reported results, greatly disregarding differences that could exist in both the training and testing of models. Proper comparison studies require more than selecting comparable models over comparable tasks to allow a solid discussion. Here we present the very first steps in the consolidation of a fair experimental protocol, proposing the construction of training and testing data sets and homogenizing neural architecture performance evaluation. Moreover, we enforce the incorporation of statistical testing as part of the analysis of results following the insights proposed and detailed in [3], including such tests as part of model evaluation validates the significance of observable results. Exploring results using statistics in the particular case of NAS can highlight the advantages that accompany novel discovered architectures and the algorithms that generated them.

The following three subsections will describe the different elements that form this protocol in more detail, allowing researchers to apply it as part of their experimentation. The discussion that accompanies these sections should convince other researchers of the benefits that employing protocol can provide in their analysis of NAS-produced results.

2.1. The Data sets needed for the evaluation

The first element that defines this protocol is a collection of data sets for training, validation, and testing of automatically discovered neural architectures. This element propitiates compatibility of the protocol in various practical environments allowing researchers to contrast proposals against results found in this work and future advancements in the area. It is crucial to mention that these datasets alone do not evaluate the process of architecture search, as evaluating this process would require prohibitive amounts of computational resources. Moreso, performing that kind of evaluation would require future research to follow the same hardware specifications to apply this protocol as a point of contrast.

Ensuring that a common agreement on the tasks used during architectural search and model training exists within the NAS pipeline will allow a clearer quantization of the progression of any novel proposal that studies this particular task. Next, we present a set of key tasks for evaluating architectures following this proposal. These tasks in the form of data sets, take well-known classical super-resolution problems preferred by convention in the field and process them using various techniques.

The first two sets we describe with this work extract instances from the DIV2K [1] data set to consolidate two tasks. Using this dataset for training and validation is common in the literature, making it perfect as a baseline for ongoing research. Not only its properties have been widely studied through many works but the availability of these datasets allows easier access to the components of our protocol.

The first task we obtain from DIV2 is a proxy dedicated to the process of architecture search, which contains a smaller but representative group of images that help maintain a reduced computational cost while training super-resolution architectures. Such a proxy task should help NAS approaches to identify promising architectures, allowing a quicker identification of well-performing models. The second set introduces a full SRIR task using data augmentation techniques applied over the previous proxy task. As such this increases the total number of instances processed by models at the end stage of NAS, increasing significantly the computational cost of training, the reason why is directed to be used only during the final training and validation of any architecture found by the search.

The DIV2k data set presents a total of 800 high-quality (2k resolution) images for training and 100 for validation. To form the proxy task data set of this protocol, patches of 64 by 64 pixels are extracted from the total 900 high-resolution images resulting in a total of 522K images for training and 66,651 images for validation. We per-

form a bicubic degradation over these images using Pillow (9.4.0V)'s `PIL.Image.BICUBIC` function, which reduces image resolution by an arbitrary ratio. For the proxy task, this data set resolution is reduced by half, and used as is for model training and validation posing a $\times 2$ SRIR task. In the full task, images are also down-sampled to a third of the original resolution on the $\times 3$ task and one-fourth in the $\times 4$ task. The three resolution sets are subjected to image flipping horizontally and vertically, as well as a 90° rotation, increasing the amount of data instances and adding more variety of patterns given to models during training in the final evaluation.

Once architectures are found and fully trained, computing their performance over unseen data is necessary to continue with the experimental contrast protocol. The data used for the test and evaluation of architectures incorporates four well-known groups of super-resolution problems, that are considered to be the norm by the literature of NAS in the context of super-resolution. These sets include Set5 [4], Set14 [25], BSD100 [19], and Urban100 [14], each containing different examples of images having different characteristics and patterns that need to be reconstructed. The proxy and training task sets use only the bicubic down-sample instances for each group, keeping things within the same domain. The data set composition is described as follows:

- Set5 is a classical data set and only contains five test images. These images comprise a baby, a bird, a butterfly, a head, and a woman.
- Set14 consists of 14 images of varying subjects, which include animals with complex patterns, people, food, scenery, and text.
- BSD100 presents another classical data set containing 100 test images ranging from images of nature to object-specific, including plants, people, and food.
- Urban100 is a more recent data set introduced by Huang et al. This set comprises 100 images focusing on human-made structures and urban scenes.

Thus, the entire set for evaluation comprises 219 high and low-resolution images, each representing a super-resolution image restoration problem. Models that wish to use this protocol need to be evaluated on the $\times 2, \times 3$, and $\times 4$ tasks for each data set, meaning that each model reconstructs a total of 657 high-resolution images with increasing degrees of degradation. These experiments should allow us not only to identify the best-performing state-of-the-art models and their characteristics but to asses in reality how different a model is from its counterparts at different levels of task complexity. This comparison scheme aims to highlight the impact that the search space design and the objectives guiding the search have on the optimal candidate architectures.

2.2. Performance measures while training and contrasting architectures

Image quality takes a decisive role in NAS not only as a way to train found models into generating visually acceptable results in super-resolution images but to guide the architectural search toward novel well-performing structures. There is a lot of work trying to discern the best way to measure crucial visual information as part of image quality, and even more attempts at trying to capture important human perception information. These attempts seek to tackle different applications of image restoration tasks, from improving the performance of machine learning approaches by restoring data that was lost due to noise [23], to producing images that cater better to our visual perception [15]. This resulted in the definition and application of many techniques assessing the quality of images, examples of which are: MSE (Mean Square Error), UIQI (Universal Image Quality Index), PSNR (Peak Signal to Noise Ratio), SSIM (Structured Similarity Index Method), HVS (Human Vision System) and FSIM (Feature Similarity Index Method).

The entirety of measures presented above in conjunction with other model-related measures such as loss variation during the architectural search, the memory footprint of architectures, prediction latency, and robustness, provide different insights that allow comparing different approaches based solely on the evaluation. Nonetheless, generally speaking, two metrics have dominated the mainstream advances in NAS for SRIR, those being PSNR and SSIM, the former used more commonly during search than the latter. Here and in later sections we focus on the usage of PSNR and the total amount of trainable parameters required by a model as ways to quantify the overall quality of found architectures and the resulting models.

On the effectiveness side, we focus on using PSNR given the strong correlation between it and SSIM as shown by Horé and Ziou in [12]. This measure evaluates the differences between a high-resolution image and the predicted super-resolution image as decibels, with a higher ratio representing a better prediction. On the efficiency side, calculating the total amount of trainable parameters of an architecture provides some insights into the complexity of a model, with deeper and wider architectures requiring more parameters. The amount of trainable parameters that result from constructing a model based on a particular architecture becomes relevant as it can be decisive in its deployment under hardware limitations.

2.3. Bayesian Statistical Assessment of experimental results

Experimentally contrasting different models, especially in machine learning, implicates the desire to identify whether or not a particular model will be able to outperform others, how large the difference between observable

results and how significant will it be in practice [8]. It is true, that frequentist approaches for hypothesis testing compute the probability of observing the expected difference in performance between classifiers. However, the statistics obtained by these approaches do not measure the probability of a model showing better accuracy than others [3]. Moreover, the common application of these techniques follows the assumptions that p-values are estimators of the probability of observing the null hypothesis and that statistical significance is the same in practice, both assumptions being incorrect. Approaches based on Bayesian statistics demonstrate to naturally provide posterior probabilities that align with the information researchers expect when performing these tests.

To further study any obtained results and assess their practical significance, Benavoli et al. present in [3], the use of a Bayesian signed-rank test analysis. We incorporate this test as a final integrating element of the protocol present in this work, such that not only do we assess the relative merit of algorithms in terms of efficiency and efficacy, but also calculate how probable it is for models in practice to keep the behavior showcased on the observed empirical results. The test used as part of this evaluation protocol allows us to ascertain whether an algorithm A will perform better than an algorithm B and the probability that this case will repeat when encountering new data, as long as it is similar to the data used for testing. We compute this probability, taking into account all the resulting performance of models over each of the data sets. After applying the Bayesian test, we end up with a set of three posterior probabilities as follows:

- $P(A \ll B)$ the probability that A is practically better than B.
- $P(A = B)$ the probability that A and B are practically equivalent.
- $P(A \gg B)$ the probability that B is practically better than A.

To perform a fair statistical test, first, results need to be normalized in a way that the performance of the models is measurable as a percentage. Second, we need to establish a parameter ($rope$) to determine whether or not the performance of the two models could be considered equivalent in practice. Depending on the problem, a greater or smaller threshold may be necessary to deem model performance significantly different. In the case of this protocol, we have determined the $rope = 0.01$, as it would be sensible for any SRIR task to consider that two classifiers are practically equivalent when the mean difference of their accuracy lies below the 1%, resulting in images being very similar perceptually.

The Bayesian test follows the implementation made available as part of the Baycomp 1.0 library in [3]. The functions of this library allow practitioners to compute the posterior probabilities of two models, after being evaluated over either a single or multiple data sets. Given the structure of this protocol posterior probabilities are calculated following a signed-rank test taking into consideration the different evaluation tasks as implemented in the function `baycomp.two_on_multiple`.

3. Evaluating architectures following the protocol

Using this protocol, we evaluated the performance of six rival NAS techniques in the context of SRIR, encompassing only machine-crafted models with similar parameters. We take NAS-DIP [5] and models A and C from HNAS [11], models A and B from FALSR [6], model A from MoreMNAS [7], ESRN and ESRN-V from [21] and DLSR from [13]. These models were selected based availability of source codes, impact, and similarity in the number of trainable parameters each model possesses. The performance of each model is evaluated based on its capacity for accurately predicting super-resolution images similar to the high-resolution samples used for a baseline taking each of the test images as an independent test and the total amount of trainable parameters required by each of the found models.

We summarize results in Tables 1 to 3. For each table, we highlight the best results of models disregarding size with bold text. Furthermore, we used the number of trainable parameters as an additional measurement of architecture quality with the first and second algorithms having the smallest footprint highlighted in light gray. Results highlighted with a darker tone of gray and white text represent the best results seen on architectures that incorporated less than 400K trainable parameters. We make these distinctions as the evaluation protocol is employed to evaluate two specific objectives, performance regarding architecture effectiveness and architecture complexity regarding efficiency.

These results indicate that in 8 of the cases, ESRN achieves a closer reconstruction of the original high-resolution image. The ESRN model achieved outstanding performance in three sets for the $\times 2$ task, three data sets on the $\times 3$, and two on the $\times 4$ tasks. Achieving such results suggests that ESRN has better capabilities for reconstructing high-resolution images concerning the other models, given that each new resolution diminishment represents a more demanding problem. While the $\times 2$ task of each set presents a challenge, the $\times 3$ and $\times 4$ tasks of each data set present more significant challenges. The difference in difficulty among each task is that a model requires achieving the same high-resolution image from smaller LR inputs each time. Moreover, each new data set after the first presents an increasing number of images that a model has to reconstruct. Each image in a data set represents a subproblem

Table 1. Performance comparison of SRIR machine-crafted architectures. This table presents the observed performance of the studied models at the super-resolution $\times 2$ task of each data set. A higher Peak-Signal-to-Noise-Ratio value represents better results. The table highlights the best results of models disregarding size with bold text. A gray background highlights architectures with the least and second least total trainable parameters. The darker gray and white text emphasizes the best result per task obtained by a model with less than 400k parameters.

Model	# of params.	PSNR on the $\times 2$ Task			
		Set5	Set14	B100	Ubran100
NAS-DIP	1,800K	35.32	31.58	29.99	29.81
FALSR-A	1,021K	37.82	33.52	32.12	31.93
FALSR-B	326K	37.61	33.29	31.97	31.28
HNAS-A	139K	37.84	33.39	32.06	31.50
HNAS-C	380K	**38.11**	33.60	32.07	31.73
MoreMNAS-A	1039K	37.63	33.23	31.95	31.24
ESRN	1039K	38.04	**33.69**	**32.23**	**32.37**
ESRN-V	324K	37.85	33.42	32.10	31.79
DLSR	322K	38.04	33.67	32.21	32.26

Table 2. Performance comparison of SRIR machine-crafted architectures. This table presents the observed performance of the studied models at the super-resolution $\times 3$ task of each data set. A higher Peak-Signal-to-Noise-Ratio value represents better results. The table highlights the best results of models disregarding size with bold text. A gray background highlights architectures with the least and second least total trainable parameters. The darker gray and white text emphasizes the best result per task obtained by a model with less than 400k parameters.

Model	# of params.	PSNR on the x3 Task			
		Set5	Set14	B100	Urban100
NAS-DIP	1,800K	30.81	27.84	26.16	26.003
FALSR-A	1,021K	32.97	29.65	28.41	28.14
FALSR-B	326K	32.80	29.34	27.88	27.31
HNAS-A	139K	32.35	29.32	27.65	27.17
HNAS-C	380K	33.01	28.52	27.30	27.10
MoreMNAS-A	1039K	32.82	29.41	27.85	27.23
ESRN	1039K	34.46	**30.43**	**29.15**	**28.42**
ESRN-V	324K	34.23	30.27	29.03	27.95
DLSR	322K	**34.49**	30.39	29.13	28.26

making B100 and Urban100 pose the biggest challenge to the models' capacity for generalization and adaptation.

To continue analyzing the results, in second place, we find the DLSR model capable of achieving the highest performance in 3 cases, one time on the $\times 3$ task and twice on the $\times 4$ task. All this while presenting a parameter reduction compared with ESRN and against the others. While ESRN has over one million learnable parameters, DLSR only contains 322K. This difference helps envision that larger, deeper, and more complex models can outperform smaller and simpler ones in most cases. Nevertheless, this also demonstrates that the quality of a model depends much on its architectural configuration. Here, a smaller model achieved comparable performance against the one found to be the best of the set, with practically a third of the parameters.

Table 3. Performance comparison of SRIR machine-crafted architectures. This table presents the observed performance of the studied models at the super-resolution $\times 4$ task of each data set. A higher Peak-Signal-to-Noise-Ratio value represents better results. The table highlights the best results of models disregarding size with bold text. A gray background highlights architectures with the least and second least total trainable parameters. The darker gray and white text emphasizes the best result per task obtained by a model with less than 400k parameters.

Model	# of params.	PSNR on the $\times 4$ Task			
		Set5	Set14	B100	Urban100
NAS-DIP	1,800K	26.41	24.59	22.42	22.28
FALSR-A	1,021K	30.33	28.21	26.11	25.36
FALSR-B	326K	28.12	26.92	23.90	23.38
HNAS-A	139K	28.22	25.43	23.40	23.27
HNAS-C	380K	28.44	25.16	24.48	23.91
MoreMNAS-A	1039K	28.72	25.87	23.93	23.40
ESRN	1039K	32.26	28.63	**27.62**	**26.24**
ESRN-V	324K	31.99	28.49	27.50	25.87
DLSR	322K	**32.33**	**28.68**	27.61	26.19

In the case of smaller models limited to 400k parameters at most, DLSR obtained the best performance generally. This model achieved the best performance in 11 out of 12 tests, only surpassed by HNAS-C once in the set5 $\times 2$ task. The DLSR employs a hierarchical and differentiable NAS method to discover and allocate cell-level blocks within an architecture based on three components: shallow residual networks, an information distillation mechanism, and a contrast-aware attention mechanism. This way, the resulting model considers a small number of parameters and operations required for its deployment while maintaining good performance. From the discussion presented in previous paragraphs and the observed results, it is possible to state that DLSR achieved the most consistent performance across the various tasks, significantly reducing the computational cost. Neural Architectures, efficient and robust, are highly desirable in the context of SRIR, even at the expense of not surpassing the best performance of the state-of-the-art.

4. Statistical analysis within the experimental protocol

The next step in the execution of this protocol serves as a way to provide more support to the discussion found in the previous section. We analyze the studied algorithms using the Bayesian signed-rank test as described in section 2.3. 4 shows if a model in a row is better ($\ll$), equal ($=$), or worse ($\gg$) than the other. Additionally, we have defined a threshold ϵ over the probabilities to determine which model is more likely to be significantly better or whether they are equivalent. For this case, we consider $\epsilon = 0.95$, as deter-

mined in [3], to ensure enough statistical significance. Elements in the table that are marked with an $*$ achieved a probability $\geq 95\%$ of being the best algorithm. This probability provides enough evidence to confirm that the model performed better, worse, or equally than the one it is compared against.

The Bayesian test validates the discussion in subsection 4.2, further reinforcing that models ESRN and DLSR are the overall best-performing of the entire group, with ESRN being the best. While ESRN and DLSR have a 70.1% probability of achieving the same practical performance, declaring ESRN as better comes from this model having a 28.9% chance of achieving a better performance over its counterpart. In the entirety of the cases, the probability of both models outperforming the others, by at least 1% more performance, surpassed 95%. However, ESRN does present approximately three times more parameters than DLSR, meaning that ESRN incorporates a costlier and more elaborate architecture. This difference in trainable parameters makes it difficult to establish the best between the two. Please refer to the supplementary material to see visual representations of the posterior probabilities calculated by the Bayesian signed-rank test.

5. Conclusions

Even in contemporary NAS research, it is clear that comparing different approaches that solve the same problem continues to pose many challenges needing addressing. Approaches that alleviate this issue are needed, not only for image restoration tasks if we wish a continual study and deployment of architecture search pipelines. We introduced an experimental protocol for NAS in the context of SRIR, allowing the possibility of comparing various super-resolution architectures more rigorously and under similar circumstances. With this protocol, we hope to motivate NAS researchers to push for more accessible and reproducible approaches. Works like this one should help bridge this methodology gap, allowing researchers to take steps toward constructing benchmarking procedures and establishing baselines expanding the range of applications for NAS.

Acknowledgments

We thank the members of the *Advanced Artificial Intelligence* group at Tecnologico de Monterrey, for providing feedback on an earlier version of this paper. The first author wants to thank the financial support given by the National Council of Science and Technology (CONACyT) under scholarship grant 829049. The research reported here was supported by CONACyT Ciencia de Frontera 2023 under grant CF-2023-I-801.

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1122–1131, 2017.

[2] Waqar Ahmad, Hazrat Ali, Zubair Shah, and Shoaib Azmat. A new generative adversarial network for medical images super resolution. 12(1):9533, 2019.

[3] Alessio Benavoli, Giorgio Corani, Janez Demšar, and Marco Zaffalon. Time for a change: a tutorial for comparing multiple classifiers through bayesian analysis. *Journal of Machine Learning Research*, 18(77):1–36, 2017.

[4] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In *23rd British Machine Vision Conference (BMVC)*, pages 135.1–135.10. BMVA press, 2012.

[5] Yun-Chun Chen, Chen Gao, Esther Robb, and Jia-Bin Huang. Nas-dip: Learning deep image prior with neural architecture search. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 442–459, Cham, 2020. Springer International Publishing.

[6] Xiangxiang Chu, Bo Zhang, Hailong Ma, Ruijun Xu, and Qingyuan Li. Fast, accurate and lightweight super-resolution with neural architecture search, 2020.

[7] Xiangxiang Chu, Bo Zhang, Ruijun Xu, and Hailong Ma. Multi-objective reinforced evolution in mobile neural architecture search, 2019.

[8] Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.*, 7:1–30, dec 2006.

[9] Thomas Elsken, Jan Hendrik Metzen, and Frank Hutter. Neural architecture search: A survey. *J. Mach. Learn. Res.*, 20(1):1997–2017, jan 2019.

[10] Edgar Galván and Peter Mooney. Neuroevolution in deep neural networks: Current trends and future challenges, 2020.

[11] Yong Guo, Yongsheng Luo, Zhenhao He, Jin Huang, and Jian Chen. Hierarchical neural architecture search for single image super-resolution. *IEEE Signal Processing Letters*, 27:1255–1259, 2020.

[12] Alain Horé and Djemel Ziou. Is there a relationship between peak-signal-to-noise ratio and structural similarity index measure? *IET Image Processing*, 7(1):12–24, 2013.

[13] Han Huang, Li Shen, Chaoyang He, Weisheng Dong, Haozhi Huang, and Guangming Shi. Lightweight image super-resolution with hierarchical and differentiable neural architecture search, 2021.

[14] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5197–5206, 2015.

[15] Shizuo Kaji and Satoshi Kida. Overview of image-to-image translation by use of deep neural networks: denoising, super-resolution, modality conversion, and reconstruction in medical imaging. 12(3):235–248, 2019.

Table 4. Bayesian analysis comparing the performance of different models. Here $\ll$ represents that the model of a specific row for any practical consideration performs better than the one on a particular column. Opposite to this, $\gg$ represents that the model on a row demonstrated worst performance than the one on the column.

Models	NAS-DIP	FALSR-A	FALSR-B	HNAS-A	HNAS-C	MoreMNAS-A	ESRN	ESRN-V	DLSR
NAS-DIP	-	$\gg *$	$\gg *$	$\gg *$	$\gg *$	$\gg *$	$\gg *$	$\gg *$	$\gg *$
FALSR-A	$\ll *$	-	$\ll *$	$\ll *$	$\ll$	$\ll *$	$\gg *$	$\gg *$	$\gg *$
FALSR-B	$\ll *$	$\gg *$	-	$\ll$	$\gg$	$\ll$	$\gg *$	$\gg *$	$\gg *$
HNAS-A	$\ll *$	$\gg *$	$\gg$	-	$\gg$	$\gg$	$\gg *$	$\gg *$	$\gg *$
HNAS-C	$\ll *$	$\gg$	$\ll$	$\ll$	-	$\ll$	$\gg *$	$\gg *$	$\gg *$
MoreMNAS-A	$\ll *$	$\gg *$	$\gg$	$\ll$	$\gg$	-	$\gg *$	$\gg *$	$\gg *$
ESRN	$\ll *$	$\ll *$	$\ll *$	$\ll *$	$\ll *$	$\ll *$	-	$\ll *$	$=$
ESRN-V	$\ll *$	$\ll *$	$\ll *$	$\ll *$	$\ll *$	$\ll *$	$\gg *$	-	$\gg *$
DLSR	$\ll *$	$\ll *$	$\ll *$	$\ll *$	$\ll *$	$\ll *$	$=$	$\ll *$	-

[16] Juncheng Li, Zehua Pei, and Tieyong Zeng. From beginner to master: A survey for deep learning-based single-image super-resolution. *arXiv preprint arXiv:2109.14335*, 2021.

[17] Marius Lindauer and Frank Hutter. Best practices for scientific research on neural architecture search. *Journal of Machine Learning Research*, 21(243):1–18, 2020.

[18] Luis Luévano García, Leonardo Chang, Heydi Vazquez, Yoanna Martínez-Díaz, and Miguel Gonzalez-Mendoza. A study on the performance of unconstrained very low resolution face recognition: Analyzing current trends and new research directions. *IEEE Access*, PP:1–1, 05 2021.

[19] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 2, pages 416–423 vol.2, 2001.

[20] Yash Mehta, Colin White, Arber Zela, Arjun Krishnakumar, Guri Zabergja, Shakiba Moradian, Mahmoud Safari, Kaicheng Yu, and Frank Hutter. Nas-bench-suite: Nas evaluation is now surprisingly easy. *arXiv preprint arXiv:2201.13396*, 2022.

[21] Dehua Song, Chang Xu, Xu Jia, Yiyi Chen, Chunjing Xu, and Yunhe Wang. Efficient residual dense block search for image super-resolution, 2019.

[22] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. *International Journal of Computer Vision*, 128(7):1867–1888, Mar 2020.

[23] Zhihao Wang, Jian Chen, and Steven C. H. Hoi. Deep learning for image super-resolution: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3365–3387, 2021.

[24] Martin Wistuba, Ambrish Rawat, and Tejaswini Pedapati. A survey on neural architecture search. *arXiv preprint arXiv:1905.01392*, 2019.

[25] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In Jean-Daniel Boissonnat, Patrick Chenin, Albert Cohen, Christian Gout, Tom Lyche, Marie-Laurence Mazure, and Larry Schumaker, editors, *Curves and Surfaces*, pages 711–730, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.