Assessing Perceptual Noise Level Defined by Human Calibration and Image Rulers

Lisa Lei

***Abstract*—We propose a framework for noise-related perceptual quality assessment and examine one of its immediate application: accessing the MRI perceptual noise level during a scan to stop it when the image is good enough. A convolutional neural network is trained to map an image to a perceptual score. The label score for training is a statistical estimation of error standard deviation calibrated with radiologist inputs. Image rulers for different scan types are used in the inference phase to determine a flexible classification threshold.**

I. Introduction

THE signal to noise ratio (SNR) is one of the major aspects of evaluating quality of most medical images. Perceptual SNR, while not well-defined, is more relevant in most cases than measurements or approximations of quantities like SNR, noise variation, and signal energy. The well-defined quantities vaguely correlate with a perceptual SNR. From a radiologist's point of view, a perceptual SNR highly dependents on the image content and the diagnosis task. Automatically accessing the perceptual SNR of images is useful for optimizing the factors affecting the SNR of medical images. Some of the factors are: reconstruction method, acquisition method, radiation dosage, etc. The same amount of noise has different effects on the diagnostic or perceptual quality of scans with different fat suppression, contrast, and field of view. We aim to access the perceptual SNR of a specific type of images and establish a framework that is generalizable to others. In this paper we work with one type of MRI scan and apply our method to solve a clinial problem associated with it.

Other things equal, the SNR of magnetic resonance images (MRIs) increases with the number of averages of repetitively acquired data. Oversampling is practiced to achieve higher SNR and better diagnostic quality, especially for relatively fast scans on static body parts. Given the tradeoff between scan time and image quality, we want to find the optimal point to stop a scan. Normally, the number of averages is set before a scan or fixed for all scans. We propose using a neural network (NN) to access a perceptual noise level from radiologists' view and make real-time decisions on when to stop the scan. This saves total scan time and decreases the number of undesired scan outcomes.

We use a perceptual standard set consistently across all types of scans by a radiologist to calibrate a quantitative noise estimation. A NN 'perceives' images and learn from the calibrated scores which better correlate with the perceptual noise level. For the inference phase, we propose using adjustable pass/fail thresholds defined by image rulers (Fig. 1). The user can pick a desired perceptual noise level for each scan by choosing a sample in the image ruler of the corresponding scan type. The input the NN is an image and the output is a predicted score that reflects the perceptual SNR of the input image.

Related works. Previous statistical no-reference noise estimations [1]–[3] are content independent and not particularly perceptual. They are patch based, assumes white Guassian noise and works in some image transfered domain. Iterative estimation in discrete cosine transform (DCT) domain (IEDD) [1] analyzes patches of a image in DCT domain to get an estimation of the noise variance. [2] selects weak-textured patches and estimates their noise variance using principal component analysis (PCA). [3] also explores the relationship between PCA outputs and noise level. [4] uses an non-convolutional NN to analyze complex image patch in the singular value decomposition (SVD) domain. [5] presented a way of defining image quality: in terms of the performance of some human or model "observer" on some task of practical interest. This way of accessing the perceptual quality is too labor heavy for our labeling task. [6] is the most recent work on general image quality, aiming to serve as an standard evaluation metric for the popular field of generative models. But it mostly focuses on how real a generated image is.

II. Methods

Our approach to the problem utilizes a supervised NN. For the general purpose of perceptual SNR evaluation, we incorporate a moderate amount of human input into the training supervision by calibrating some kind of statistical approximation of the noise level. Then for applications of achieving a specified perceptual goal, we use an image ruler in the inference phase to transfer human perception to model interpretable numbers.

A. Approximate perceptual labels by human calibration

For the training inputs, we generate m_t noisier versions of each original MRI slice by adding zero-mean white Gaussian noises (WGNs) to the k-space (i.e. measurements in Fourier domain), according to the ideal model of MRI noise [7]. The standard deviation (std) of the added noise to each version decrease with the version number $h \in \mathbb{Z}$.

For the training labels, we start with two calculated quantities (Q) as the initial approximation of the perceptual noise level. One is the pixel averaged SNR in dB, based on the difference between each noise-injected image and its original version, and SNRs for all original versions are artificially set to be 5 dB higher than that of the next cleanest version. The other one is the estimated noise std by the IEDD method [1] based on each single image.

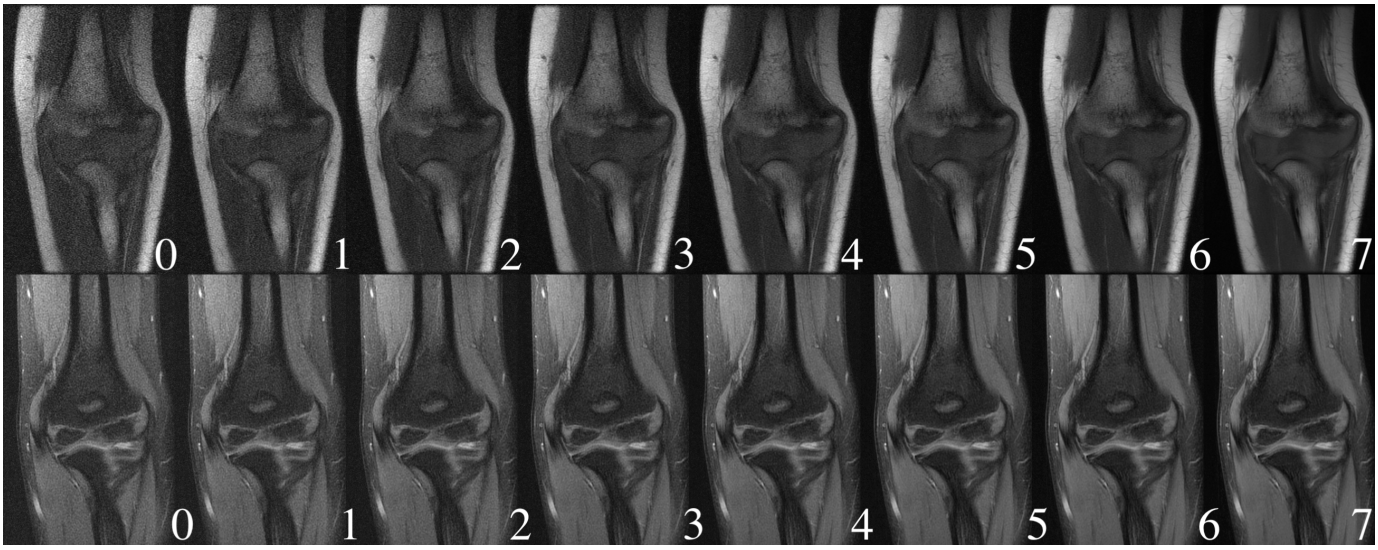

Fig. 1. Two image rulers: the top one is for non-FS scans and the bottom one is for FS scans.

To incorporate radiologist inputs in the loop and have a set of training labels that better represents the perceptual quality of image from radiologists' view, we collect labels from a radiologist to roughly calibrate Q. For every slice i, a radiologist selects one version $h_i \in [1,\ m_t]$, or something better than the cleanest version available ($h_i = m_t + 1$), that meets the least satisfactory perceptual noise level for that specific slice. The selection/labeling is done using the interface shown in Fig. 7. The selected images have almost equivalent perceptual SNR by our definition so we set their calibrated scores to one constant μ and adjust Q of the other versions within one slice accordingly. We calibrate/align calculated quantities of all images as follows:

$$y_i^v = Q(x_i^v), \tag{1}$$

$$\mu = \frac{1}{n}\sum_{i=1}^{n} y_i^{h_i}, \tag{2}$$

$$\hat{y}_i^v = y_i^v + \mu - y_i^{h_i},\ \left(h_i \le m_t\right) \tag{3}$$

$$\hat{y}_i^v = y_i^v + \mu - 2y_i^{m_t} + y_i^{m_t-1},\ \left(h_i = m_t + 1\right) \tag{4}$$

where y_i^v is the original calculated quantity for ith slice vth version, and $\hat{y}$ are the calibrated scores.

The alignment does not guarantee accurate labels for all images in the training set, only for the selected images. But this saves a great amount of labeling effort comparing to labeling all images with a vaguely defined perceptual score or aligning all images by their perceptual quality. Besides, after training on a large enough dataset, the small discrepancy in specific training labels does not bias the overall learning outcome. And for most applications we do not need a score that is consistent across all images. For example, for optimizing the reconstruction method for a image, we only need a perceptual score that is consistent among versions of that one image. For the applications with scan-specific image rulers, as the one we present later, we only need a perceptual score that is consistent among a scan type.

Convolutional NNs are competent at capturing perceptual quality [8]. Our model utilizes a pyramidal CNN to map images to scalar scores. The training objective is the root-mean-square error (RMSE) between the model output $D(x) \in \mathbb{R}$ and the label $y \in \mathbb{R}$:

$$\min_{\theta_D} \sqrt{\sum_{i}^{n} \frac{(D(x_i;\theta_D) - y_i)^2}{n}}\ . \tag{5}$$

B. Inference with image ruler

The desired SNR varies with the anatomy in concern so we need an adjustable target or threshold. It is hard for a human user to indicate a desired perceptual quality by abstract numbers. So in the inference phase, we present an image ruler: m_r versions of a sample slice with decreasing noise level (Fig. 1). The user has the option to pick a desired noise level for the upcoming scan by selecting one version, or in between two versions, in the ruler. The score given by the model D on the chosen slice is used as the threshold score (see eq. (6)). This is motivated by the assumption that our trained CNN outputs similar value for similar-looking images.

Then we introduce multiple scan-specific rulers, where the sample slice comes from a previous scan of the same contrast, or/and anatomy, etc. The more rulers we use, the more similar they can be to each scan, and the easier it is for the user to relate to the upcoming scan. Using multiple scan-specific rulers partially addresses the content dependency of perceptual SNR, and our model is only required to output scores that is consistent within similar scans grouped by the rulers.

As new measurements keep coming, images are reconstructed then passed to the model every few seconds until their scores $D(x;\theta_D)$ reach the threshold extracted from the corresponding image ruler, when the scan is terminated. The threshold is given by:

$$\frac{D(u_r^i;\theta_D) + D(u_r^j;\theta_D)}{2}, \tag{6}$$

where u_r^i is the ith version in the rth ruler, $i, j \in \{0,\ 1,\ \ldots\ m_r - 1\}$ and $|i - j| = \{0, 1\}$.

C. Non-deep-learning baseline

We use a support vector machine (SVM) with a 3-degree polynomial kernel and L2 regularization [9] to perform binary classification on a fixed threshold. The training label is given by:

$$y_i^v = \mathbb{1}\{v \geq h_i\}. \tag{7}$$

In the inference phase, the SVM outputs 0 or 1. This baseline achieves the same end goal as one specific application and threshold setting of our model. The binary decision boundary is fixed according to the training set and cannot adjust to any specific requirements. It cannot be used to compare images or be extended to other applications. The task is easier because the model is directly trained on binary labels equivalent to those for the end test. On the other hand, our proposed model learns a mapping to continuous scores which are later indirectly converted to serve the end binary test.

III. Experiments

Dataset. The raw data is from the Stanford Lucile Packard Children's Hospital (LPCH) and extracted from our private database. The multi-coil fully-sampled k-space data is from 2D fast-spin echo scans of knee and elbow. We use sum-of-squares reconstruction (i.e., $image = \sqrt{\sum_{i=1}^{c} |\mathcal{F}^{-1}(k_i)|^2}$) and interpolate the images to the 512×512 standard size. To simulate noisy versions of a slice, we add white zero-mean Gaussian noise to the original k-space data k_i before performing the same reconstruction. We simulate four noisier versions for each slice in the training set (i.e. $m_t = 5$) by adding independent WGNs with four incremental stds to the real and imaginary parts of the k-space data. Fig. 7 shows one slice in a subject where the rightmost image is the reconstructed original image and the left four are its noise-injected versions. We simulate seven noisier versions of the two slices for the image rulers (i.e. $m_r = 8$) the same way.

There are 1250 unique slices from 180 subjects in the training set and 91 unique slices (two versions/images each) from 20 subjects in the test set. One radiologist from LPCH labeled the whole training set; two radiologists from LPCH labeled the test set. We have 300 selection labels for the training set and the label for each slice is set to that for its closest labeled slice. We include one noise-injected version with each original slice in the test set to balance the pass and fail classes. Images in the test set are matched to versions in its corresponding image ruler that appears most similarly noisy. We use two image rulers: one for fat-suppressed scans (FS) and one for non-FS scans. 11, 15, 18, 20, 21, 17, 27, 53 of the test images are marked as 0-7, respectively. Under this relatively even distribution of image qualities, the binary classification on any threshold is not a trivial task.

Model tuning. First, we tried three training objectives: absolute difference, RMSE and MSE. Second, we searched for the layer and feature map numbers for a fully convolutional network. Third, we tried adding a fully-connected (FC) layer to the end of the CNN. Fourth, we tried four dropout rates at the last conv layer: 0.0, 0.15, 0.3, 0.4. Fifth, we tried a learning rate of $10^{-4}, 5 \times 10^{-5}, 10^{-5}$ and picked 10^{-4}. Since noise is a lower level feature, we tried adding a loss from the second conv layer output followed by a FC layer to inject gradient to the lower layers. We tried a feature map size of 5x5 and 3x3. Finally, we tried leaky ReLU instead of ReLU. The model is shown in Fig. 2. We use a mini-batch size of 10, consisting of 2 slices $\times$ 5 versions. For the SVM model, we search for the best kernel among: polynomial, sigmoid, Gaussian, and linear. The image inputs to the SVM are resized to 128×128 by 4×4 average pooling with a stride of 4.

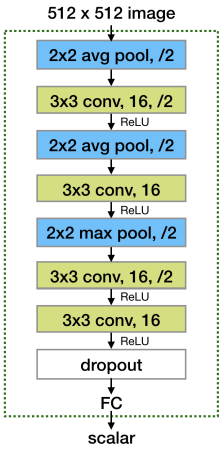

Fig. 2. Model architecture.

We train our CNN model with four sets of labels: SNR, calibrated SNR (cSNR), IEDD, and calibrated IEDD (cIEDD). Classification accuracies on the default threshold (in between 4 and 5 in rulers) from the four models compared with the SVM baseline and three previous methods [1]–[3] are shown in Table I. The calibrated models are better than their counterparts. Our proposed model – CNN trained with cIEDD label – achieves the best performance under the image-ruler-defined-thresholds setting; its confusion matrix is shown in Fig. 3. Test accuracies achieved by one fixed threshold that best classifies the test set itself are included in Table I. Despite not generalizable to other dataset, some of the accuracies are significantly lower than that with our ruler-defined thresholds. The test method degenerates the problem to binary classification so that our results seem less impressive. But we keep our approach flexible and generalizable. Although the SVM is showing good accuracy, note its approach is simpler as explained before.

Fig. 4 shows two false negative test examples from the proposed model. Fig. 5 shows two false positive test examples from the same model. These incorrectly classified examples are close to the threshold. Fig. 6 shows two test examples overlaid with their saliency map. The model seems to focus on the edges of the objects, which is similar to human roughly accessing the overall clarity of an image.

We examine the robustness of trained models to various thresholds on the ruler. Table II shows test accuracies from the same models on three thresholds. The NN models perform best

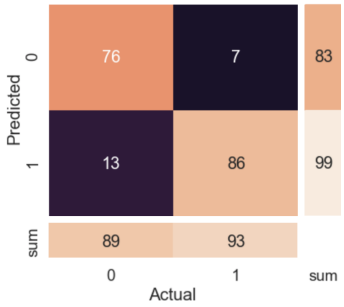

Fig. 3. Confusion matrix of our proposed method with 89.01% test accuracy.

on the default threshold used for validation and calibration. The accuracy drop on other standards is not severe from the calibrated model, and the default is the most commonly used.

Since the test set is labeled differently than the training set and we do not have a ground-truth perceptual score, the training and test error cannot be compared. It is desirable to have a fair way of evaluating the model output scores directly besides evaluating the accuracy after the discretize-to-binary step.

The inference time for a 10-image batch is 0.03 seconds on a NVIDIA TITAN Xp GPU [10].

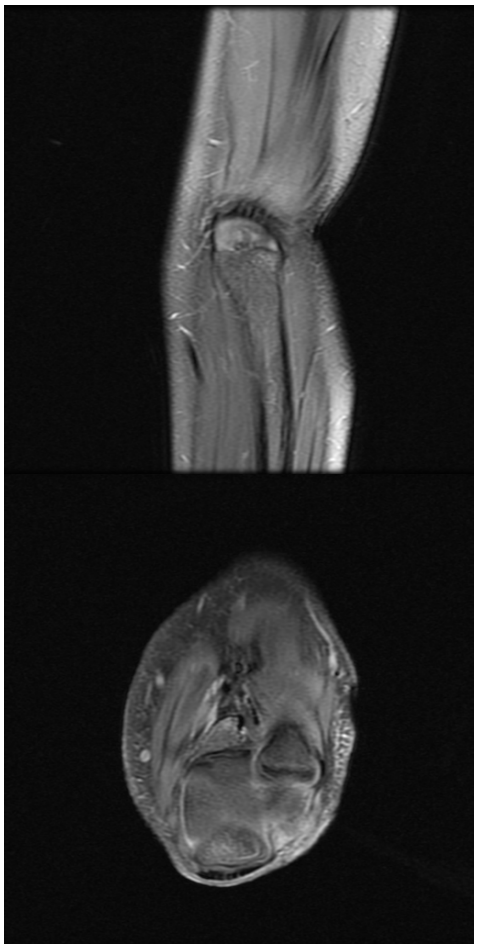

Fig. 4. Two false negative example.

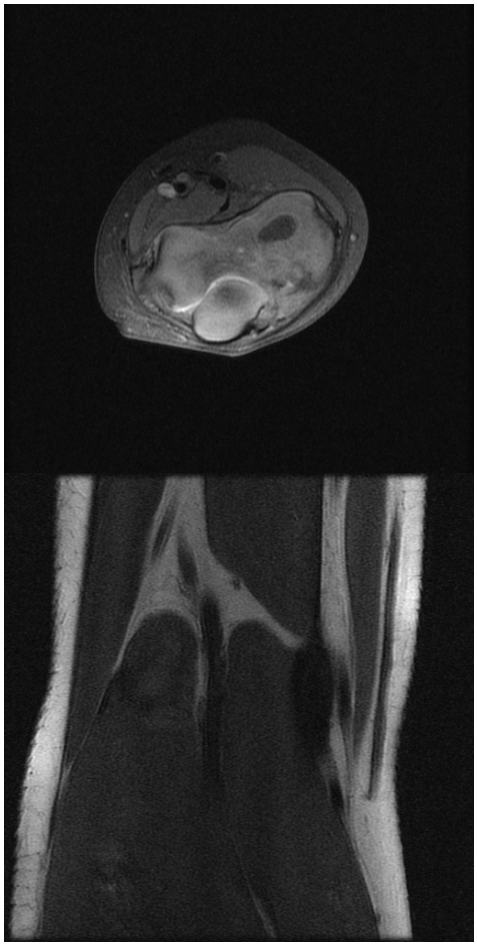

Fig. 5. Two false positive example.

IV. Conclusion

We introduced a CNN model to estimate the perceptual noise level of an image. One of its application is making real-time decisions to stop a scan right after a desired noise level is obtained. The desired noise level can be tailored at inference time by referring to image rulers similar to the upcoming scan. Both the proposed human calibration for training label and the ruler-defined thresholds for inference contribute significantly to the classification accuracy.

Next, we will expand the dataset to include more anatomies, try adding more types of rulers, and deploy the model for clinical use. We have 462 more subjects from 8 more anatomies to be labeled. The same framework can be used for tuning the regularization parameter for compressed-sensing reconstruction.

The code is at https://stanford.box.com/s/9x684hk42hwkgw8gx8i2iu76gv7vryoo

References

[1] M. Ponomarenko, N. V. Gapon, V. V. Voronin, and K. O. Egiazarian, "Blind estimation of white gaussian noise variance in highly textured images," *CoRR*, vol. abs/1711.10792, 2017. [Online]. Available: http://arxiv.org/abs/1711.10792

[2] X. Liu, M. Tanaka, and M. Okutomi, "Noise level estimation using weak textured patches of a single noisy image," in *2012 19th IEEE International Conference on Image Processing*, Sep. 2012, pp. 665–668.

TABLE I

TEST ACCURACIES OF THREE PREVIOUS METHODS AND OUR NN MODELS TRAINED WITH FOUR SETS OF LABELS. INFERENCE ON THE TEST SET IS PERFORMED WITH THE TWO RULER-DEFINED FLEXIBLE THRESHOLDS (OPTIMIZED ON A SMALL VALIDATION SET) AND A SINGLE BEST THRESHOLD (OPTIMIZED FOR THE EXACT SAME TEST SET).

Threshold type	Chen [3]	Liu [2]	IEDD [1]	SVM	NN - SNR	NN - cSNR	NN - IEDD	NN - cIEDD
Ruler defined	71.98%	74.73%	79.67%	NA	80.21%	82.97%	86.26%	**89.01%**
Single best	61.53%	68.13%	69.78%	88.5%	81.87%	82.97%	80.77%	87.91%

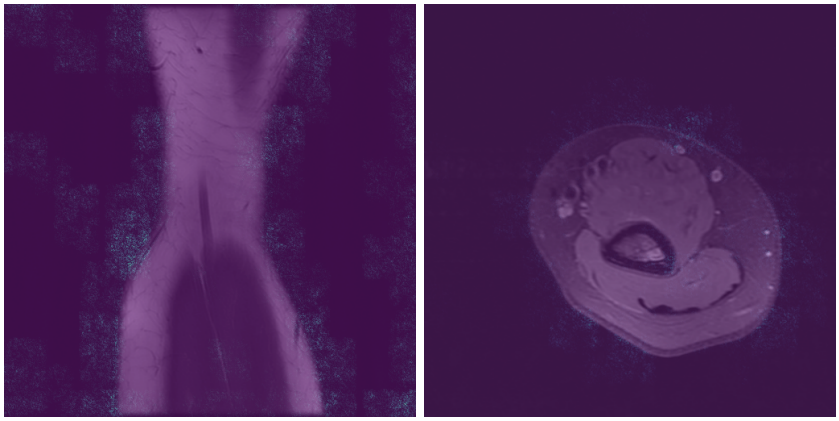

Fig. 6. Two test images overlaid with their saliency maps.

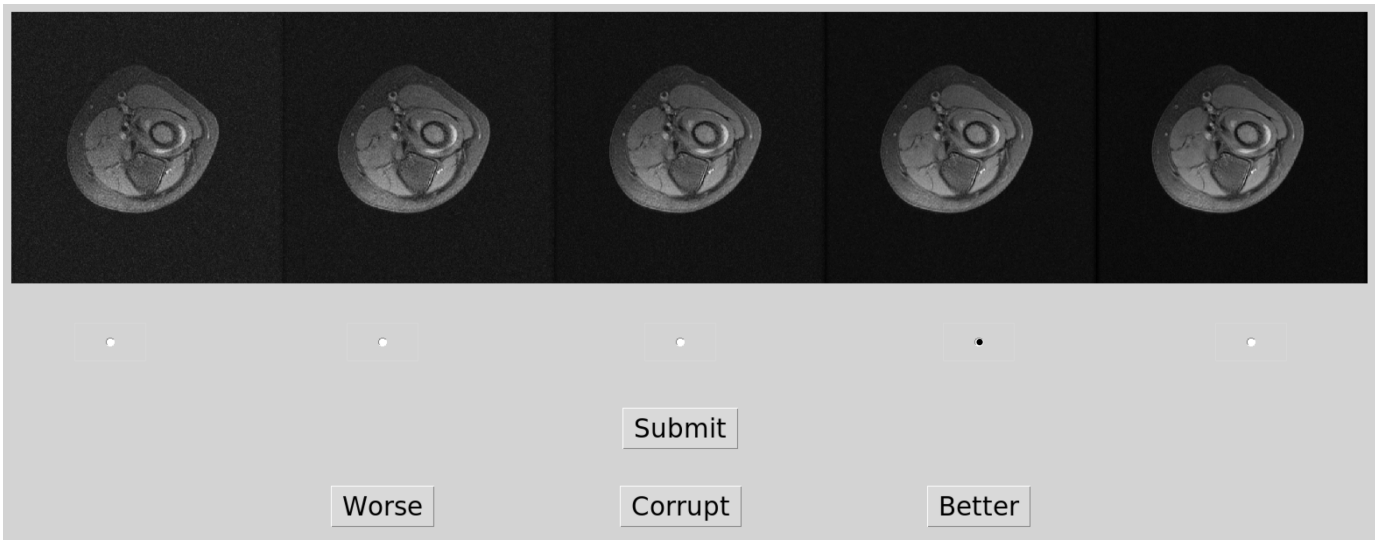

Fig. 7. The graphic interface for training set labeling. Standard deviations of the added noises decreases to 0 from left to right.

TABLE II

TEST ACCURACIES WHEN THE MINIMAL DESIRED STANDARD IS SET BETWEEN 3 AND 4 (I.E., 3 | 4), 4 AND 5, 5 AND 6 IN THE RULER, FROM TRAINED MODEL VALIDATED ON THE DEFAULT STANDARD (BETWEEN 4 AND 5).

Threshold	IEDD	NN - IEDD	NN - cIEDD
3 | 4	78.02%	81.32%	88.46%
4 | 5	79.67%	86.26%	89.01%
5 | 6	81.32%	82.42%	85.16%

[3] G. Chen, F. Zhu, and P. A. Heng, "An efficient statistical method for image noise level estimation," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec 2015, pp. 477–485.

[4] E. Turajlic, A. Begović, and N. Škaljo, "Application of artificial neural network for image noise level estimation in the svd domain," *Electronics*, vol. 8, no. 2, p. 163, 2019.

[5] H. H. Barrett, J. Yao, J. P. Rolland, and K. J. Myers, "Model observers for assessment of image quality," *Proceedings of the National Academy of Sciences*, vol. 90, no. 21, pp. 9758–9765, 1993. [Online]. Available: https://www.pnas.org/content/90/21/9758

[6] S. Zhou, M. Gordon, R. Krishna, A. Narcomey, L. F. Fei-Fei, and M. Bernstein, "Hype: A benchmark for human eye perceptual evaluation of generative models," in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 3444–3456. [Online]. Available: http://papers.nips.cc/paper/8605-hype-a-benchmark-for-human-eye-perceptual-evaluation-of-generative-models.pdf

[7] S. Aja-Fernández and G. Vegas-Sánchez-Ferrero, *Statistical Noise Models for MRI*. Cham: Springer International Publishing, 2016, pp. 31–71.

[8] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-

time style transfer and super-resolution," in *European Conference on Computer Vision*, 2016.

[9] scikitlearn. (2019) Support vector machines. [Online]. Available: https://scikit-learn.org/stable/modules/svm.html

[10] NVIDIA. (2019) Titan xp graphics card with pascal architecture. [Online]. Available: https://www.nvidia.com/en-us/titan/titan-xp/