Fidelity Estimation Improves Noisy-Image Classification with Pretrained Networks

Xiaoyu Lin, Deblina Bhattacharjee, Majed El Helou, *Members, IEEE*, and Sabine Süsstrunk, *Fellow, IEEE*.

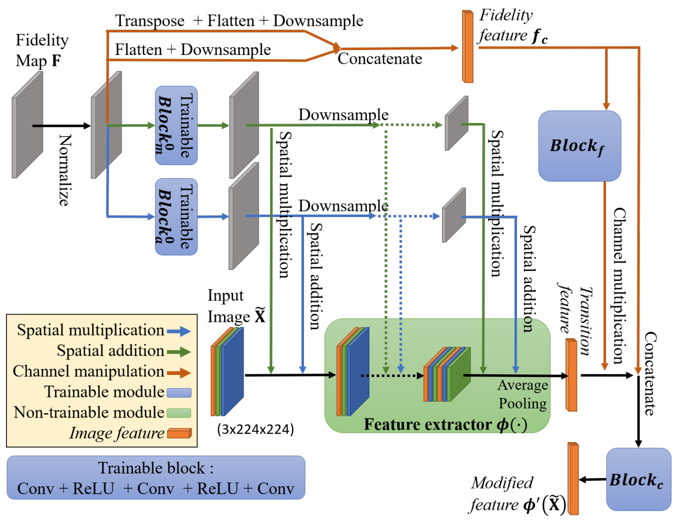

Fig. 3. Detailed module architecture (without ensemble) of our proposed method based on ResNet-50 [1]. FC, Conv and ReLU indicate one fully connected layer, one convolutional layer and a rectified linear unit activation, respectively. The pretrained module (feature extractor) is shown in green, and the modules containing trainable parameters are shown in blue.

TABLE III
CLASSIFICATION ACCURACY (%) OF THE BASELINE AND OUR FG-NIC METHODS (BASED ON RESNET-50) ON CALTECH-101. THE BEST AND SECOND BEST RESULTS *aside from the oracle configuration* ARE IN BOLD AND UNDERLINED RESPECTIVELY.

Methods	Experimental setup	Uniform degradation (σ) 0.1	0.2	0.3	0.4	0.5	Varying 1D	2D
Pretrained	Test on noisy	83.87	56.95	27.13	10.95	5.50	54.23	50.18
	Test on restored	**92.03**	84.83	70.16	50.17	31.18	81.38	80.17
Retrain on noisy	Test on noisy	89.62	87.92	86.23	84.50	82.04	87.59	87.38
	Test on restored	89.85	87.90	82.16	73.98	62.19	85.86	85.68
Retrain on restored	Test on noisy	84.18	64.52	36.82	15.53	5.85	59.62	56.74
	Test on restored	91.89	**89.86**	**87.84**	**85.39**	**82.64**	**89.49**	**89.11**
FG-NIC (Pretrained)	Single	89.85	87.76	84.68	80.49	75.39	86.42	86.64
	Ensemble	91.79	89.08	84.72	78.81	70.80	87.71	87.60
FG-NIC (Oracle)	Single	*90.91*	*89.46*	*87.75*	*86.07*	*83.83*	*89.02*	*88.89*
	Ensemble	*91.97*	*90.29*	*88.02*	*85.82*	*82.87*	*89.61*	*89.53*

I. EXTENDED EXPERIMENTAL RESULTS

The Caltech-101 [2] dataset is similar to Caltech-256 with fewer images and categories. We select 30 images per class as the training set and keep the same procedure as for Caltech-256. The results are consistent and given in Table III.

II. COMPUTATIONAL COMPLEXITY

We show in Table IV the number of Multiply–accumulate operation (MAC) and trainable parameters used in each network or module, highlighting the computational efficiency of our proposed approach.

TABLE IV
#MACS IS THE NUMBER OF MULTIPLY–ACCUMULATE OPERATIONS (IN BILLION), AND #PARAMS IS THE NUMBER OF TRAINABLE PARAMETERS (IN MILLION).

Network model	#MACs (G)	#Params (M)
ResNet-50 (Classification)	4.11	24.03
Ensemble module	4.11	0.21
Our FG-NIC	0.08	10.49

III. ABLATION STUDY

We run a series of experiments on Caltech-256 and conduct an in-depth analysis of the results to show the improvement of each module in our proposed method. For the end-to-end setup, we train our FG-NIC and fidelity map estimator together. For the different fidelity map inputs and outputs, downsampling methods, and generally the ablation studies, we use the oracle setup with the single model. The results are given in Table V and support the statements and design decisions we make in our main manuscript.

TABLE V
IN-DEPTH ANALYSIS AND ABLATION STUDY RESULTS (CLASSIFICATION ACCURACY IN %). BOLD NUMBERS SHOW THE BEST ACCURACY AND UNDERLINED NUMBERS SHOW THE NEXT BEST ACCURACY.

Methods		Uniform degradation (σ) 0.1	0.2	0.3	0.4	0.5
Our FG-NIC: oracle + single		80.45	78.14	**76.13**	74.06	71.75
Our FG-NIC: end-to-end + single		80.57	75.21	68.22	60.01	51.03
Pretrained on clean test on restored		77.99	67.06	53.00	38.08	25.53
Fidelity map input: restored		80.41	78.15	**76.13**	**74.12**	**71.77**
Fidelity map output: ℓ_2 distance		80.46	78.13	76.08	73.90	71.55
Fidelity map output: cosine		79.45	75.07	69.27	62.56	55.18
Downsampling: bicubic		**80.69**	**78.25**	76.09	73.81	71.43
Downsampling: nearest neighbor		80.32	78.01	75.81	73.65	71.37
Ablation study	w/o spatial multiplication	80.17	77.40	74.89	72.34	69.49
	w/o spatial addition	80.09	76.93	73.82	70.47	67.08
	w/o channel multiplication	80.32	77.72	75.70	73.51	70.95
	w/o channel concatenation	80.55	77.55	74.68	71.77	68.63

REFERENCES

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[2] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 594–611, 2006.