Distilling the Knowledge from Normalizing Flows

Dmitry Baranchuk [1 2] **Vladimir Aliev** [1] **Artem Babenko** [1 2]

Abstract

Normalizing flows are a powerful class of generative models demonstrating strong performance in several speech and vision problems. In contrast to other generative models, normalizing flows have tractable likelihoods and allow for stable training. However, they have to be carefully designed to represent invertible functions with efficient Jacobian determinant calculation. In practice, these requirements lead to overparameterized and sophisticated architectures that are inferior to alternative feed-forward models in terms of inference time and memory consumption. In this work, we investigate whether one can distill knowledge from flow-based models to more efficient alternatives. We provide a positive answer to this question by proposing a simple distillation approach and demonstrating its effectiveness on state-of-the-art conditional flow-based models for image super-resolution and speech synthesis.

1. Introduction

Normalizing flows (NF) (Dinh et al., 2015; Rezende & Mohamed, 2015) are a class of generative models that construct complex probability distributions by applying a series of invertible transformations to a simple base density (typically multivariate Gaussian). Such a design allows for exact likelihood computation via change-of-variables formula. Therefore, NF can be straightforwardly trained via likelihood maximization. This NF property is appealing for practitioners since alternative GAN-based models require adversarial optimization, which suffers from vanishing gradients, mode collapse, oscillating or cyclic behavior (Goodfellow, 2016).

To enable exact likelihood computation, NF architectures must be composed of invertible modules that also support the efficient calculation of their Jacobian determinant. A large number of such modules have been recently developed, including autoregressive, bipartite, linear and residual transformations (Huang et al., 2018; Kingma et al., 2016; Papamakarios et al., 2017; van den Berg et al., 2018; Durkan et al., 2019; Dinh et al., 2017; 2015; Rezende & Mohamed, 2015; Kingma & Dhariwal, 2018; Chen et al., 2019). While some modules are significantly more efficient than others, normalizing flows are generally inferior to feed-forward counterparts (e.g., GANs) in terms of sampling time. In particular, autoregressive flows (Huang et al., 2018; Papamakarios et al., 2017) use a slow sequential generation procedure, while bipartite flows (Dinh et al., 2017; Kingma & Dhariwal, 2018) can require a lot of submodules with low expressive power. Moreover, invertibility limits the size of the inner representation leading to impractically deep models. The more comprehensive discussion of the background and related work is deferred to Appendix A.

However, in many applications, one does not need explicit density estimation but requires efficient inference at deployment. This raises a question whether one can relinquish the invertibility of normalizing flows to improve their runtime and memory consumption after training. In this work, we investigate whether this can be achieved through the knowledge distillation approach. In particular, we describe how to distill knowledge from pretrained flow-based models into efficient architectures, which do not suffer from NF design limitations. Our code and models are available online.[1]

We summarize our contributions as follows:

- We propose a plain training strategy and student design that allow for knowledge distillation from conditional normalizing flows to feed-forward models with streamlined inference and lower memory consumption. To the best of our knowledge, this is the first work that proposes the NF distillation to feed-forward architectures.
- In our experiments, we empirically confirm the effectiveness of our method on the state-of-the-art flow-based models for super-resolution (SRFlow (Lugmayr et al., 2020a)) and speech synthesis (WaveGlow (Prenger et al., 2018)). We can achieve up to $\times 10$ speedups with no perceptible loss in quality.

[1]Yandex, Moscow, Russia [2]National Research University Higher School of Economics, Russia. Correspondence to: Dmitry Baranchuk <dmitry.baranchuk@graphics.cs.msu.ru>.

Third workshop on *Invertible Neural Networks, Normalizing Flows, and Explicit Likelihood Models* (ICML 2021). Copyright 2021 by the author(s).

[1]https://github.com/yandex-research/distill-nf

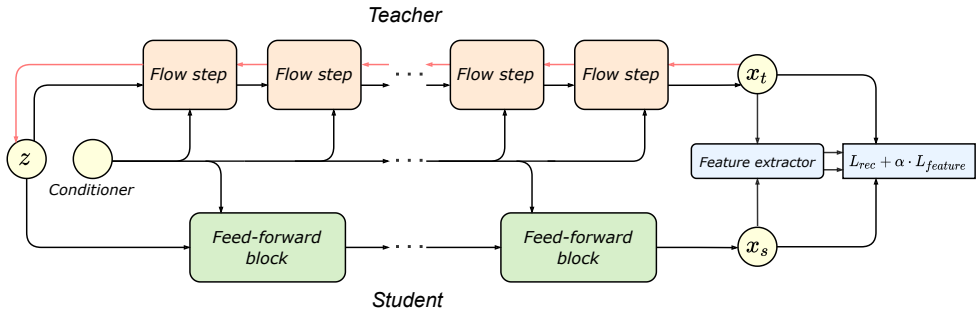

Figure 1. Overall scheme of the proposed knowledge distillation approach. Knowledge is transferred from a flow-based teacher to a feed-forward student using reconstruction and feature losses. As a result, a student is no longer restricted to be a flow and hence can exploit more efficient architectures.

2. Method

This section formally describes the details of our distillation approach. We also provide a general guide how one can design a student architecture for a given flow-based teacher. As illustrative examples, we use two state-of-the-art models from speech and vision domains.

2.1. Distillation procedure

A pretrained conditional flow-based model defines a deterministic bijective mapping from a noise sample $z \sim N(0, \sigma I)$ and contextual information c to an output sample x. Our distillation approach approximates this mapping by a more efficient model in a supervised manner. The general scheme of our approach is presented in Figure 1.

2.2. Objective

The proposed distillation objective is a combination of *reconstruction* and *feature* losses:

$$L = L_{rec} + \alpha \cdot L_{feature} \tag{1}$$

where α is a hyperparameter to balance the loss terms.

For both applications, the reconstruction loss is an average L_1-norm between the student and teacher samples:

$$L_{rec} = \|T_\theta(z, c) - S_\psi(z, c)\|_1 \tag{2}$$

where T_θ and S_ψ correspond to the teacher and student models, respectively.

The feature loss for SR model distillation is a perceptual distance between generated images computed by LPIPS (Zhang et al., 2018b). A pretrained VGG16 model (Simonyan & Zisserman, 2015) is used for LPIPS calculation.

The feature loss for speech vocoder distillation is a *multi-resolution $STFT$[2] loss* (van den Oord et al., 2018; Yamamoto et al., 2020; 2019), which is a sum of $STFT$ losses with different parameters (i.e., FFT size, window size and frame shift). In more detail, a single $STFT$ loss is the sum of the *spectral convergence* and *log $STFT$ magnitude* terms (Yamamoto et al., 2020).

[2] $STFT$ stands for the Short-Term Fourier Transform

2.3. Student design

The proposed student design can be described by a general recipe: "just clone a teacher architecture and take advantage of not being a flow anymore". In other words, our student models inherit the reduced teacher architecture and get free of the following constrains:

Invertibility. In flow-based models, the size of the inner representations cannot exceed the dimensions of input data vectors. This restriction might lead to much deeper and hence inefficient models. Since the student no longer has to be reversible, one can vary representation dimensions arbitrarily. As a result, student architectures with much fewer blocks can achieve similar performance.

Tractable Jacobian determinants. Flow-based models usually have to exploit specific operations for easy-to-compute Jacobian determinants. Therefore, the student models might benefit from replacing these operations with more efficient and expressive ones.

Below, we describe the particular design options on the example of flow-based models with affine coupling layers (Dinh et al., 2017; Kingma & Dhariwal, 2018). The supportive illustration is provided in Figure 2.

a) The initial flow-based model with affine coupling layers, e.g., SRFlow or WaveGlow with fewer parameters.

b) One can consider adjusting the inner representation dimensions to increase the overall expressive power.

c) Affine coupling layers split the input z into two halves z_a and z_b. Then, z_b is used to predict the affine transform parameters to process z_a. This split operation is needed to make the Jacobian determinant easily computable. In the student model, one can remove the split operation and transform the entire input vector.

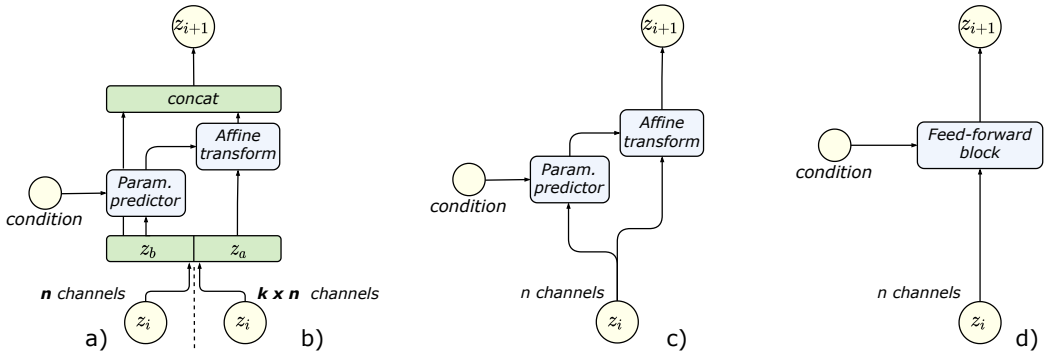

Figure 2. Student design options on the example of affine coupling layers. a) Affine coupling block; b) the same but with increased inner representation dimensions; c) affine block without partition operations; d) the affine block is substituted by a feed-forward module.

d) Finally, it might be helpful to substitute the entire affine block with a commonly used feed-forward module for the corresponding task.

In our SRFlow and WaveGlow students, we adhere to the options b) and d). Specifically, we tune the inner representation size and replace the flow steps with stacked RRDBs (Wang et al., 2018) and WaveNet blocks (Oord et al., 2016), respectively. The detailed descriptions of the student architectures are provided in Appendix B. The student design ablation on the example of WaveGlow is presented in Appendix F.

3. Experiments

In this section, we compare the learned student against the corresponding teachers and other relevant baselines from the literature.

3.1. Super-resolution

For super-resolution, we compare the student to the following set of models:

- **ESRGAN**(Wang et al., 2018), **RankSRGAN** (Mittal et al., 2013) — the recent GAN-based super-resolution models. ESRGAN is trained by a combination of reconstruction, perceptual and adversarial losses. RankSRGAN uses the additional ranker network to optimize non-differentiable perceptual metrics.

- **RRDB** (Wang et al., 2018) — the same as ESRGAN but trained only with L_1 objective.

- **SRFlow** (Lugmayr et al., 2020a) — a flow-based teacher model. The model is optimized for efficient inference by precomputing inversions for 1×1 invertible convolutions and fusing actnorm layers.

- **SRFlow Student** — a proposed student model designed according to Section 2.3.

Datasets. The evaluation is performed on the DIV2K dataset (Agustsson & Timofte, 2017) — one of the established benchmarks for single image super-resolution. In addition to the DIV2K train set, the Flickr2K dataset (Lim et al., 2017) is also used for training. We consider $\times4$ and $\times8$ scaling factors between LR and HR images.

Evaluation metrics. As the primary measure of perceptual performance, we report the established LPIPS (Zhang et al., 2018b), which is proven to correlate with human estimation (Lugmayr et al., 2020b). We also report the standard fidelity-oriented metrics, Peak Signal to Noise Ratio (PSNR) and structural similarity index (SSIM) (Wang et al., 2004). In addition, we evaluate consistency with the LR image by reporting the LR-PSNR, computed as PSNR between the downsampled SR image and the original LR image.

To evaluate diversity, we follow (Mao et al., 2019) and measure the average pairwise LPIPS distances between the generated samples for a particular LR image. Larger pairwise distances indicate higher diversity.

Evaluation results. We report metrics as well as corresponding inference time and a number of parameters in Table 1. For most baseline models, we reproduce the results on publicly released checkpoints. Runtimes are measured on a single Tesla V100 on a single validation sample in half precision. The SRFlow student achieves the same metric values as the teacher being by $\times2.1$ and $\times4.9$ faster for $\times4$ and $\times8$ scaling factors, respectively.

Also, we provide qualitative results for $\times4$ and $\times8$ scaling factors in the appendix, see Figure 4 and Figure 5, respectively. The inference for the teacher and student models is performed for the same input noise vectors z. We observe that the student produces samples with quality similar to the teacher.

Note that the student preserves the teacher diversity, which indicates that our distillation procedure is potentially able to produce student models with exploitable latent space (Lugmayr et al., 2020a).

Models ×4	PSNR ↑	SSIM ↑	LPIPS ↓	LR-PSNR ↑	Param. (M)	Time (ms)	Diversity↑
RRDB	29.43	0.84	0.253	49.04	16.70	303 ± 1	—
RankSRGAN	26.55	0.75	0.132	42.35	1.554	40 ± 1	—
ESRGAN	26.63	0.76	0.115	42.39	16.70	303 ± 1	—
SRFlow	**27.07**	**0.76**	**0.120**	**49.75**	**39.54**	**866 ± 2**	**0.068**
SRFlow Student	**27.32**	**0.77**	**0.120**	**49.52**	**20.39**	**420 ± 1**	**0.064**
Models ×8	PSNR ↑	SSIM ↑	LPIPS ↓	LR-PSNR ↑	Param. (M)	Time (ms)	Diversity↑
RRDB	25.54	0.70	0.418	44.98	16.74	110 ± 2	—
RankSRGAN	—	—	—	—	—	—	—
ESRGAN*	22.18	0.58	0.277	31.35	16.74	110 ± 2	—
SRFlow	**23.01**	**0.57**	**0.271**	**50.20**	**50.84**	**690 ± 5**	**0.168**
SRFlow Student	**23.30**	**0.58**	**0.269**	**48.36**	**16.16**	**140 ± 2**	**0.159**

Table 1. Evaluation metrics on the DIV2K dataset. LPIPS is considered as a primary measure of perceptual quality. The student models provide similar metrics being by $\times 2.1$ and $\times 4.9$ faster for $\times 4$ and $\times 8$ scaling factors, respectively. (*) denotes that metrics are taken from (Lugmayr et al., 2020a).

3.2. Speech Synthesis

Here, we consider the following set of models for speech synthesis evaluation:

- **WaveGlow** (Prenger et al., 2018) — a flow-based teacher model. For efficient inference, we remove weight norms (Salimans & Kingma, 2016) and precompute inversions for convolutional layers.
- **NanoFlow** (Lee et al., 2020) — a recent flow-based vocoder, which produces samples with comparable quality to WaveGlow, but has about $\times 30$ fewer parameters.
- **MelGAN** (Kumar et al., 2019), **Parallel WaveGAN** (Yamamoto et al., 2020), **HiFi-GAN** (Kong et al., 2020) — recent GAN-based vocoders. While MelGAN and ParallelWaveGAN are inferior to WaveGlow, HiFi-GAN represents the current state-of-the-art.
- **WG Student** — a student model designed according to Section 2.3. For evaluation, we consider three configurations with 4/4/2 WaveNet blocks of 128/96/96 hidden channels and denote them as V1/V2/V3, respectively.

Datasets. All experiments are performed on the LJ speech dataset (Ito & Johnson, 2017), which is one of the most common benchmarks in speech synthesis. We use a sampling rate of $22,050$kHz and produce mel-spectrograms of the original audio according to (Prenger et al., 2018). While Prenger et al. (2018) originally provide train/validation/test splits for the LJ speech dataset, other methods are not always consistent with them. Therefore, for a fair comparison of the pretrained models, we collect a novel evaluation set and provide its details in Appendix E.

Evaluation results. We compare vocoders in the setting where models are conditioned on ground-truth mel-spectrograms as in previous works (Prenger et al., 2018; Kim et al., 2019). For all baseline models, we use officially released pretrained models. As a primary metric for generated audio evaluation, we report Mean Opinion Score (MOS) with corresponding 95 confidence intervals (CI) in Table 2. In Appendix D, we describe the detailed protocol used for MOS evaluation. The inference speed is reported in MHz, which stands for 10^6 audio samples per second. The runtimes are measured on a single Tesla V100 in half precision with sufficiently large batch sizes and sequence lengths to suppress non-model related overheads.

Models	MOS	Param. (M)	Speed (MHz)
Ground-truth	4.50 ± 0.06	—	—
NanoFlow	3.67 ± 0.09	2.82	0.369
WaveGlow	**3.92 ± 0.08**	**87.73**	**1.26**
WG Student V1	3.91 ± 0.08	9.53	9.36
WG Student V2	**3.89 ± 0.09**	**6.35**	**12.21**
WG Student V3	3.62 ± 0.09	3.18	23.62
Parallel WaveGAN	3.84 ± 0.09	1.44	2.76
MelGAN	3.58 ± 0.08	4.27	28.40
HiFi-GAN	**4.10 ± 0.08**	**1.46**	**54.13**

Table 2. Evaluation results on the LJ Speech dataset. WG Student V1 and V2 demonstrate similar speech quality to the WaveGlow teacher being by $\times 7.4$ and $\times 9.7$ faster, respectively.

V1 and V2 students demonstrate speech quality similar to the teacher and provide $\times 7.4$ and $\times 9.7$ faster speech generation, respectively. Moreover, these students have $\times 9.2$ and $\times 13.8$ fewer parameters compared to the teacher.

4. Conclusion

In this work, we address the problem of high computational cost of normalizing flows and explain how one can increase the efficiency by giving up invertibility and tractable Jacobian, which are often not necessary for deployed models. In particular, we describe an effective knowledge distillation method from a flow-based teacher to more lightweight student architectures. We empirically demonstrate that the models distilled from normalizing flows are an appealing combination of simplicity, stability and efficiency, needed for typical production pipelines.

References

Agustsson, E. and Timofte, R. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017.

Arık, S. Ö., Chrzanowski, M., Coates, A., Diamos, G., Gibiansky, A., Kang, Y., Li, X., Miller, J., Ng, A., Raiman, J., Sengupta, S., and Shoeybi, M. Deep voice: Real-time neural text-to-speech. In Precup, D. and Teh, Y. W. (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 195–204, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR.

Behrmann, J., Grathwohl, W., Chen, R. T. Q., Duvenaud, D., and Jacobsen, J.-H. Invertible residual networks. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 573–582, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

Chen, N., Zhang, Y., Zen, H., Weiss, R. J., Norouzi, M., and Chan, W. Wavegrad: Estimating gradients for waveform generation. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=NsMLjcFaO8O.

Chen, R. T. Q., Rubanova, Y., Bettencourt, J., and Duvenaud, D. K. Neural ordinary differential equations. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 31, pp. 6571–6583. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/file/69386f6bb1dfed68692a24c8686939b9-Paper.pdf.

Chen, R. T. Q., Behrmann, J., Duvenaud, D., and Jacobsen, J. Residual flows for invertible generative modeling. In *Advances in Neural Information Processing Systems*, 2019.

Dinh, L., Krueger, D., and Bengio, Y. Nice: Non-linear independent components estimation. *CoRR*, abs/1410.8516, 2015.

Dinh, L., Sohl-Dickstein, J., and Bengio, S. Density estimation using real NVP. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

Durkan, C., Bekasov, A., Murray, I., and Papamakarios, G. Neural spline flows. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32, pp. 7511–7522. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/7ac71d433f282034e088473244df8c02-Paper.pdf.

Goodfellow, I. Nips 2016 tutorial: Generative adversarial networks. *Advances in Neural Information Processing Systems*, 2016.

Grathwohl, W., Chen, R. T. Q., Bettencourt, J., and Duvenaud, D. Scalable reversible generative models with free-form continuous dynamics. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=rJxgknCcK7.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. URL https://arxiv.org/abs/1503.02531v1.

Ho, J., Chen, X., Srinivas, A., Duan, Y., and Abbeel, P. Flow++: Improving flow-based generative models with variational dequantization and architecture design. In *International Conference on Machine Learning*, pp. 2722–2730. PMLR, 2019.

Huang, C.-W., Krueger, D., Lacoste, A., and Courville, A. Neural autoregressive flows. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2078–2087, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR.

Ito, K. and Johnson, L. The lj speech dataset. https://keithito.com/LJ-Speech-Dataset/, 2017.

Kalchbrenner, N., Elsen, E., Simonyan, K., Noury, S., Casagrande, N., Lockhart, E., Stimberg, F., van den Oord, A., Dieleman, S., and Kavukcuoglu, K. Efficient neural audio synthesis. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2410–2419, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR.

Karami, M., Schuurmans, D., Sohl-Dickstein, J., Dinh, L., and Duckworth, D. Invertible convolutional flow. In *NeurIPS*, 2019.

Kim, S., Lee, S.-G., Song, J., Kim, J., and Yoon, S. FloWaveNet : A generative flow for raw audio. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*,

pp. 3370–3378. PMLR, 09–15 Jun 2019. URL http://proceedings.mlr.press/v97/kim19b.html.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kingma, D. P. and Dhariwal, P. Glow: Generative flow with invertible 1x1 convolutions. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2018.

Kingma, D. P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I., and Welling, M. Improved variational inference with inverse autoregressive flow. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 29, pp. 4743–4751. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/ddeebdeefdb7e7e7a697e1c3e3d8ef54-Paper.pdf.

Kong, J., Kim, J., and Bae, J. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. F., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 17022–17033. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/c5d736809766d46260d816d8dbc9eb44-Paper.pdf.

Kong, Z., Ping, W., Huang, J., Zhao, K., and Catanzaro, B. Diffwave: A versatile diffusion model for audio synthesis. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=a-xFK8Ymz5J.

Kumar, K., Kumar, R., de Boissiere, T., Gestin, L., Teoh, W. Z., Sotelo, J., de Brébisson, A., Bengio, Y., and Courville, A. C. Melgan: Generative adversarial networks for conditional waveform synthesis. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32, pp. 14910–14921. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/6804c9bca0a615bdb9374d00a9fcba59-Paper.pdf.

Ledig, C., Theis, L., Huszár, F., Caballero, J., Aitken, A., Tejani, A., Totz, J., Wang, Z., and Shi, W. Photo-realistic single image super-resolution using a generative adversarial network. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 105–114, 2017.

Lee, S., Kim, S., and Yoon, S. Nanoflow: Scalable normalizing flows with sublinear parameter complexity, 2020.

Lim, B., Son, S., Kim, H., Nah, S., and Lee, K. M. Enhanced deep residual networks for single image super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017.

Lugmayr, A., Danelljan, M., Gool, L. V., and Timofte, R. Srflow: Learning the super-resolution space with normalizing flow, 2020a.

Lugmayr, A., Danelljan, M., and Timofte, R. Ntire 2020 challenge on real-world image super-resolution: Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020b.

Mao, Q., Lee, H.-Y., Tseng, H.-Y., Ma, S., and Yang, M.-H. Mode seeking generative adversarial networks for diverse image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1429–1437, 2019.

Mehri, S., Kumar, K., Gulrajani, I., Kumar, R., Jain, S., Sotelo, J., Courville, A., and Bengio, Y. Samplernn: An unconditional end-to-end neural audio generation model, 2016. URL http://arxiv.org/abs/1612.07837.

Mittal, A., Soundararajan, R., and Bovik, A. C. Making a "completely blind" image quality analyzer. *IEEE Signal Processing Letters*, 20(3):209–212, 2013. doi: 10.1109/LSP.2012.2227726.

Oord, A. v. d., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., and Kavukcuoglu, K. Wavenet: A generative model for raw audio. *CoRR*, 2016. URL https://arxiv.org/abs/1609.03499.

Papamakarios, G., Pavlakou, T., and Murray, I. Masked autoregressive flow for density estimation. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30, pp. 2338–2347. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/6c1da886822c67822bcf3679d04369fa-Paper.pdf.

Ping, W., Peng, K., Gibiansky, A., Arik, S. O., Kannan, A., Narang, S., Raiman, J., and Miller, J. Deep voice 3: 2000-speaker neural text-to-speech. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=HJtEm4p6Z.

Ping, W., Peng, K., and Chen, J. Clarinet: Parallel wave generation in end-to-end text-to-speech. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=HklY120cYm.

Ping, W., Peng, K., Zhao, K., and Song, Z. WaveFlow: A compact flow-based model for raw audio. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 7706–7716, Virtual, 2020. PMLR.

Prenger, R., Valle, R., and Catanzaro, B. Waveglow: A flow-based generative network for speech synthesis, 2018.

Ren, Y., Ruan, Y., Tan, X., Qin, T., Zhao, S., Zhao, Z., and Liu, T.-Y. Fastspeech: Fast, robust and controllable text to speech. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32, pp. 3171–3180. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/f63f65b503e22cb970527f23c9ad7db1-Paper.pdf.

Ren, Y., Hu, C., Tan, X., Qin, T., Zhao, S., Zhao, Z., and Liu, T.-Y. Fastspeech 2: Fast and high-quality end-to-end text to speech, 2020.

Rezende, D. and Mohamed, S. Variational inference with normalizing flows. In Bach, F. and Blei, D. (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1530–1538. PMLR, 07–09 Jul 2015.

Ribeiro, F. P., Florêncio, D., Zhang, C., and Seltzer, M. L. Crowdmos: An approach for crowdsourcing mean opinion score studies. *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2416–2419, 2011.

Romero, A., Ballas, N., Kahou, S., Chassang, A., Gatta, C., and Bengio, Y. Fitnets: Hints for thin deep nets. *CoRR*, abs/1412.6550, 2015.

Salimans, T. and Kingma, D. P. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 29, pp. 901–909. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/ed265bc903a5a097f61d3ec064d96d2e-Paper.pdf.

Sau, B. and Balasubramanian, V. Deep model compression: Distilling knowledge from noisy teachers. 10 2016.

Serrà, J., Pascual, S., and Segura Perales, C. Blow: a single-scale hyperconditioned flow for non-parallel raw-audio voice conversion. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32, pp. 6793–6803. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/9426c311e76888b3b2368150cd05f362-Paper.pdf.

Shen, J., Pang, R., Weiss, R., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Skerrv-Ryan, R., Saurous, R., Agiomvrgiannakis, Y., and Wu, Y. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. pp. 4779–4783, 04 2018. doi: 10.1109/ICASSP.2018.8461368.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

Smith, L. N. and Topin, N. Super-convergence: Very fast training of residual networks using large learning rates, 2018. URL https://openreview.net/forum?id=H1A5ztj3b.

Uria, B., Côté, M.-A., Gregor, K., Murray, I., and Larochelle, H. Neural autoregressive distribution estimation. *The Journal of Machine Learning Research*, 17 (1):7184–7220, 2016.

van den Berg, R., Hasenclever, L., Tomczak, J., and Welling, M. Sylvester normalizing flows for variational inference. In *proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*, 2018.

van den Oord, A., Kalchbrenner, N., Espeholt, L., kavukcuoglu, k., Vinyals, O., and Graves, A. Conditional image generation with pixelcnn decoders. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 29, pp. 4790–4798. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/b1301141feffabac455e1f90a7de2054-Paper.pdf.

van den Oord, A., Li, Y., Babuschkin, I., Simonyan, K., Vinyals, O., Kavukcuoglu, K., van den Driessche, G., Lockhart, E., Cobo, L., Stimberg, F., Casagrande, N., Grewe, D., Noury, S., Dieleman, S., Elsen, E., Kalchbrenner, N., Zen, H., Graves, A., King, H., Walters, T., Belov, D., and Hassabis, D. Parallel WaveNet: Fast high-fidelity speech synthesis. In *Proceedings of the 35th International Conference on Machine Learning*, Proceedings of Machine Learning Research, Stockholmsmässan, Stockholm Sweden, 2018. PMLR.

Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y., and Change Loy, C. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, September 2018.

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. Image quality assessment: From error visibility to structural similarity. *Trans. Img. Proc.*, 13(4):600–612, April 2004. ISSN 1057-7149. doi: 10.1109/TIP.2003.819861.

Yamamoto, R., Song, E., and Kim, J.-M. Probability Density Distillation with Generative Adversarial Networks for High-Quality Parallel Waveform Generation. In *Proc. Interspeech 2019*, pp. 699–703, 2019. doi: 10.21437/Interspeech.2019-1965. URL http://dx.doi.org/10.21437/Interspeech.2019-1965.

Yamamoto, R., Song, E., and Kim, J.-M. Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram, 2020.

Yang, G., Huang, X., Hao, Z., Liu, M.-Y., Belongie, S., and Hariharan, B. Pointflow: 3d point cloud generation with continuous normalizing flows. *arXiv*, 2019.

Zhang, K., Zuo, W., and Zhang, L. Learning a single convolutional super-resolution network for multiple degradations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3262–3271, 2018a.

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018b.

Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., and Fu, Y. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 286–301, 2018c.

Zhang, Y., Tian, Y., Kong, Y., Zhong, B., and Fu, Y. Residual dense network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2472–2481, 2018d.

Appendices

A. Background & Related work

This section briefly describes the basics of normalizing flows and their applications in the image super-resolution and speech synthesis tasks. Finally, we provide a short overview of knowledge distillation techniques.

A.1. Normalizing flows

Normalizing flows rely on change-of-variable formula to compute exact log likelihood as a sum of Jacobian log-determinants. In general, computing the Jacobian determinant of a transformation with N-dimensional inputs and outputs has $O(N^3)$ computational complexity, which is intractable for large N in typical deep learning applications. Therefore, one of the challenges is to design invertible architectures with efficient determinant calculation.

Currently, there are two main families of flow-based models with easy-to-compute Jacobian determinants: based on autoregressive and bipartite transformations. The autoregressive family includes *autoregressive flow* (AF) (Papamakarios et al., 2017; Huang et al., 2018) and *inverse autoregressive flow* (IAF) (Kingma et al., 2016). AFs resemble autoregressive models (Uria et al., 2016), which allow for parallel density estimation but perform sequential synthesis. In contrast, IAF poses parallel synthesis but sequential density estimation, making likelihood-based training very slow. The second family includes *bipartite transformations* (Dinh et al., 2017; Kingma & Dhariwal, 2018), which provide both efficient likelihood-based training and parallel synthesis. However, bipartite flows are usually less expressive than the autoregressive ones and hence the models require a larger number of layers and parameters. Recent works develop more powerful and efficient normalizing flows. Ping et al. (2020); Lee et al. (2020) combine the ideas of autoregressive and bipartite transformations. Others propose more expressive invertible layers and architectures (Ho et al., 2019; Durkan et al., 2019; Karami et al., 2019; Chen et al., 2019; Behrmann et al., 2019) or continuous transformations (Grathwohl et al., 2019; Chen et al., 2018).

At the moment, conditional NFs are gaining popularity in various practical speech and vision applications. In particular, several recent works (Prenger et al., 2018; Kim et al., 2019; Lugmayr et al., 2020a; Yang et al., 2019) exploit the ideas of Glow (Kingma & Dhariwal, 2018), RealNVP (Dinh et al., 2017) and continuous flows (Chen et al., 2018) for waveform synthesis, image super-resolution, and point cloud generation. In this work, we focus on the state-of-the-art flow-based models for image super-resolution (Lugmayr et al., 2020a) and speech synthesis (Prenger et al., 2018).

A.2. Super-resolution

Super-resolution (SR) is one of the fundamental image processing problems which aims to improve the quality of low-resolution (LR) images by upscaling them to high-resolution (HR) ones with natural high-frequency details.

Single image super-resolution approaches tend to either discover the most efficient neural network architectures (Zhang et al., 2018d;c;a) or improve high-frequency details generation by introducing more sophisticated objectives (Ledig et al., 2017; Wang et al., 2018; Mittal et al., 2013). In contrast to these approaches, the SRFlow model (Lugmayr et al., 2020a) designs a flow-based architecture that is trained by likelihood maximization. As a result, SRFlow estimates a conditional distribution of natural HR images corresponding to a given LR image.

SRFlow is built upon the Glow (Kingma & Dhariwal, 2018) architecture that gradually transforms a sample from initial distribution $z \sim N(0, I)$ into the HR image. The model is conditioned on representations of LR images delivered by a deterministic encoder (Wang et al., 2018).

A.3. Speech Synthesis

The state-of-the-art performance for speech synthesis is also achieved by deep generative models. In typical speech synthesis pipelines, neural vocoders[3] play one of the most important roles. Usually, a neural vocoder synthesizes time-domain waveform and is conditioned on mel-spectrograms from a text-to-spectrogram model (Arık et al., 2017; Ping et al., 2018; Ren et al., 2019; 2020; Shen et al., 2018).

Long-standing state-of-the-art neural vocoders are autoregressive models (Oord et al., 2016; Kalchbrenner et al., 2018; Mehri et al., 2016) that provide impressive speech quality but suffer from slow sequential generation. Therefore, a wide line of works aims to speedup their inference.

Flow-based models have been successfully applied for parallel waveform synthesis with fidelity comparable to autoregressive models (van den Oord et al., 2018; Prenger et al., 2018; Ping et al., 2020; Lee et al., 2020; Kim et al., 2019; Ping et al., 2019; Serrà et al., 2019). Among flow-based vocoders, WaveGlow (Prenger et al., 2018), WaveFlow (Ping et al., 2020), NanoFlow (Lee et al., 2020) and FloWaveNet (Kim et al., 2019) have a particular emphasis on efficiency. In this work, we specifically consider the WaveGlow model (Prenger et al., 2018) — one of the current state-of-the-art flow-based vocoders.

Being efficient enough, GAN-based vocoders (Kumar et al., 2019; Yamamoto et al., 2020) used to provide inferior speech fidelity compared to the state-of-the-art flow-based

[3] vocoder — speech waveform synthesizer

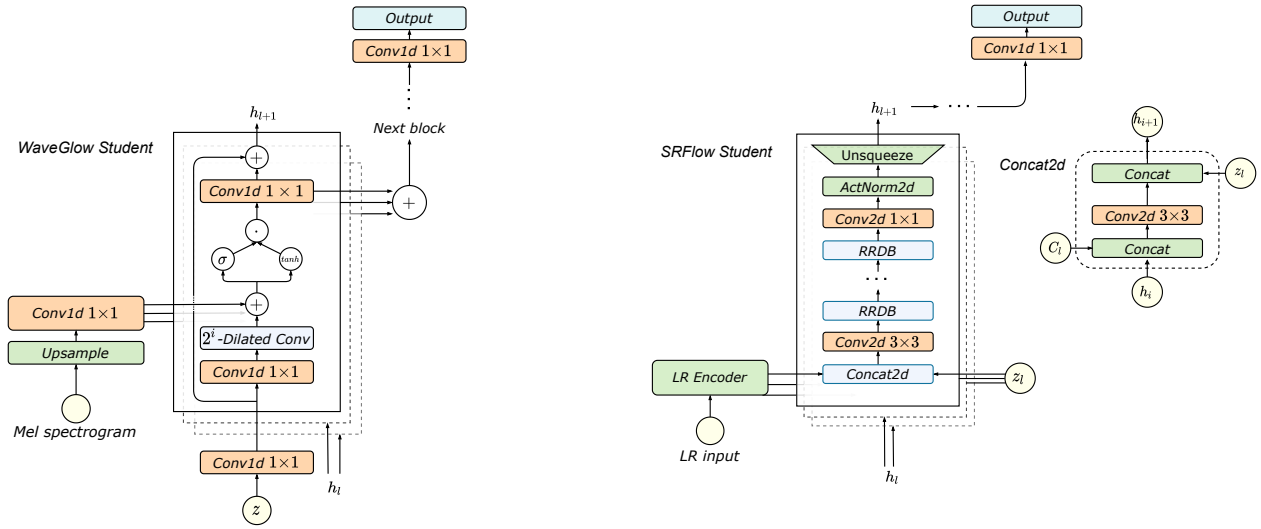

Figure 3. **Left:** WaveGlow student is presented as a sequence of non-causal WaveNet blocks (Oord et al., 2016), which are conditioned on upsampled mel-spectograms. **Right:** SRFlow student consists of L levels. At each level l, feature maps from the previous level are first combined with corresponding LR encoding c_l and noise vector z_l. Then, stacked RRDBs followed by an unsqueeze operation are applied.

models. However, recently proposed HiFi-GAN (Kong et al., 2020) efficiently delivers high-quality speech due to the training procedure with multi-scale (Kumar et al., 2019) and multi-period discriminators (Kong et al., 2020).

An alternative line of works proposes vocoders based on the denoising diffusion framework (Kong et al., 2021; Chen et al., 2021). These models demonstrate promising speech quality but are not efficient enough for many usage cases.

A.4. Knowledge Distillation

Knowledge distillation is one of the most popular compression and acceleration techniques for large models and ensembles of neural networks (Hinton et al., 2015; Sau & Balasubramanian, 2016; Romero et al., 2015). Specifically, the idea of knowledge distillation is to train a single efficient student on predictions produced by a computationally expensive teacher.

The closest works to ours distill knowledge from an expensive autoregressive neural vocoder to normalizing flows with parallel inference. In particular, Parallel WaveNet (van den Oord et al., 2018) and ClariNet (Ping et al., 2019) transfer knowledge from a pretrained WaveNet (Oord et al., 2016) to IAF (Kingma et al., 2016). As opposed to these works, we address the computational inefficiency of flow-based models and propose to transfer knowledge from normalizing flows to feed-forward networks.

B. Student architectures

Below, we describe the architectures for WaveGlow and SRFlow students in more detail and depict them in Figure 3 (Left) and (Right), respectively.

B.1. SRFlow Student

LR encoder. Both teacher and student models condition on the LR representations produced by the LR encoder. The LR encoder is a feed-forward SR architecture based on 23 Residual-in-Residual Dense Blocks (RRDB) (Wang et al., 2018). Originally, RRDBs are composed of 3 Residual Dense Blocks (RDB) and each RDB consists of 5 convolutional layers. In our student models, we vary the numbers of RDB blocks as well as convolutional layers within them.

The LR encoder is unchanged for $\times 4$ scaling factor. For $\times 8$ scaling factor, the RRDB module is substituted by a single RDB with 3 convolutional layers.

Base network. The base architecture consists of L levels. At each level l, the model combines the corresponding LR representation C_l, the activations from the previous level h_l and the noise vector z_l by the `Concat2d` layer. Here, one can vary h channel dimensions, according to Section 2.3 b).

Then, instead of a sequence of flow steps in the teacher model, stacked RRDB modules follow. For $\times 4$ scaling factor, we use 6 RRDB modules of 2 RDBs with 3 convolutional layers. For $\times 8$, a single original RRDB is used.

Finally, similarly to the SRFlow teacher, the transition step and unsqueeze operation are applied. The transition step is Conv1×1 followed by the Actnorm2d layer (Kingma & Dhariwal, 2018). The unsqueeze operation doubles the spatial resolution of feature maps and reduces their channel dimension by a factor of 4.

B.2. WaveGlow Student

WaveGlow student represents a sequence of conditional non-causal WaveNet blocks (Oord et al., 2016). We start and finish with Conv1×1 layers to adjust the input channel dimension. Each WaveNet block is organized into 8 residual layers, each consisting of a dilated convolution followed by a gated activation unit (van den Oord et al., 2016) and 1×1 convolution. The upsampled mel-spectrograms are added to the intermediate activations before the gated activation unit. In contrast to the teacher, a student does not inject noise between intermediate WaveNet blocks but obtains the entire z at the very beginning.

C. Training details

SRFlow students. The models are trained on patches of size 128×128 by the Adam optimizer (Kingma & Ba, 2014) with learning rate of $2e{-}4$ for 100 epochs. The learning rate is dropped by a factor of 2 at each 25-th epoch. The batch sizes of 80 and 128 are used for 4 and 8 scaling factors, respectively. The loss coefficient α is 10.

WaveGlow students. The models are optimized on short segments of length 16000 samples by the Adam optimizer (Kingma & Ba, 2014) with a learning rate of $1e{-}4$ for 200000 iterations. For faster convergence, we use One Cycle learning rate schedule (Smith & Topin, 2018) with 5000 warm up steps. The batch size is 256. The loss coefficient α is $1e{-}2$ for most student settings.

All hyperparameters are tuned on the hold-out train samples, which do not participate in the final evaluation. The training is performed on 8× Tesla V100.

D. Mean Opinion Score

We crowd-sourced MOS tests on 400 English-speaking individuals per model. Each rater had to pass training on 21 validation audio samples from various models with golden results and corresponding clarifications. Then, they listened to 10 audio samples from a single model and rated their naturalness on a five-point scale, see Table 3. Each rater was asked to wear headphones and work in a quiet environment. To aggregate MOS results for each model, we follow the protocol from (Kumar et al., 2019).

E. Evaluation set for Speech Synthesis

The collected set consists of 50 utterances trimmed from publicly available recordings by Linda Johnson from LibriVox. We carefully choose recordings to be similar to the original dataset and normalize audio samples accordingly. This test set will be publicly available online for reproducibility purposes.

Rating	Quality
5	Natural speech. Distortions are imperceptible.
4	Mostly natural speech. Distortions are slightly perceptible and rare.
3	Equally natural and unnatural speech. Distortions are perceptible and almost permanent.
2	Mostly unnatural speech. Distortions are annoying, but not objectionable.
1	Completely unnatural speech. Distortions are very objectionable.

Table 3. The rating scale presented to 400 individuals for MOS evaluation. This rating scale was adapted from (Ribeiro et al., 2011) to better fit modern speech synthesis applications.

Models	MOS	Param. (M)	Speed (MHz)
Flow Student	3.42 ± 0.08	6.28	12.25
Wide Flow Student	**3.93 ± 0.08**	6.37	11.95
Affine Student	**3.90 ± 0.09**	6.28	12.32
WG Student	**3.89 ± 0.09**	6.35	12.21

Table 4. The comparison of the student design principles described in Section 2.3. All options provide essentially the same performance. This indicates that relaxing either invertibility or Jacobian constraints is sufficient to achieve significant improvements.

F. Student design ablation

To identify the core change to the architecture that improves the student performance, we ablate the student design principles described in Section 2.3 on the speech synthesis task. In this experiment, we consider the student models with 4 WaveNet blocks of 96 channels and evaluate the following configurations in correspondence with Figure 2:

a) **Flow Student** — a student corresponds to the smaller WaveGlow model with all NF restrictions mentioned in Section 2.3.

b) **Wide Flow Student** — a flow student where the inner representation channel dimension is increased from 8 to 96. Note that this model is not invertible anymore.

c) **Affine Student** — the same as a flow student but the partition operations are removed, see Figure 2c). This student is still invertible but has $O(N^3)$ complexity of Jacobian determinant calculation.

d) **WG Student** — a student model where affine coupling layers are replaced with WaveNet blocks.

According to Table 4, once any of the NF constraints are lifted, the corresponding student can achieve significantly better speech quality compared to the flow student of the same capacity.

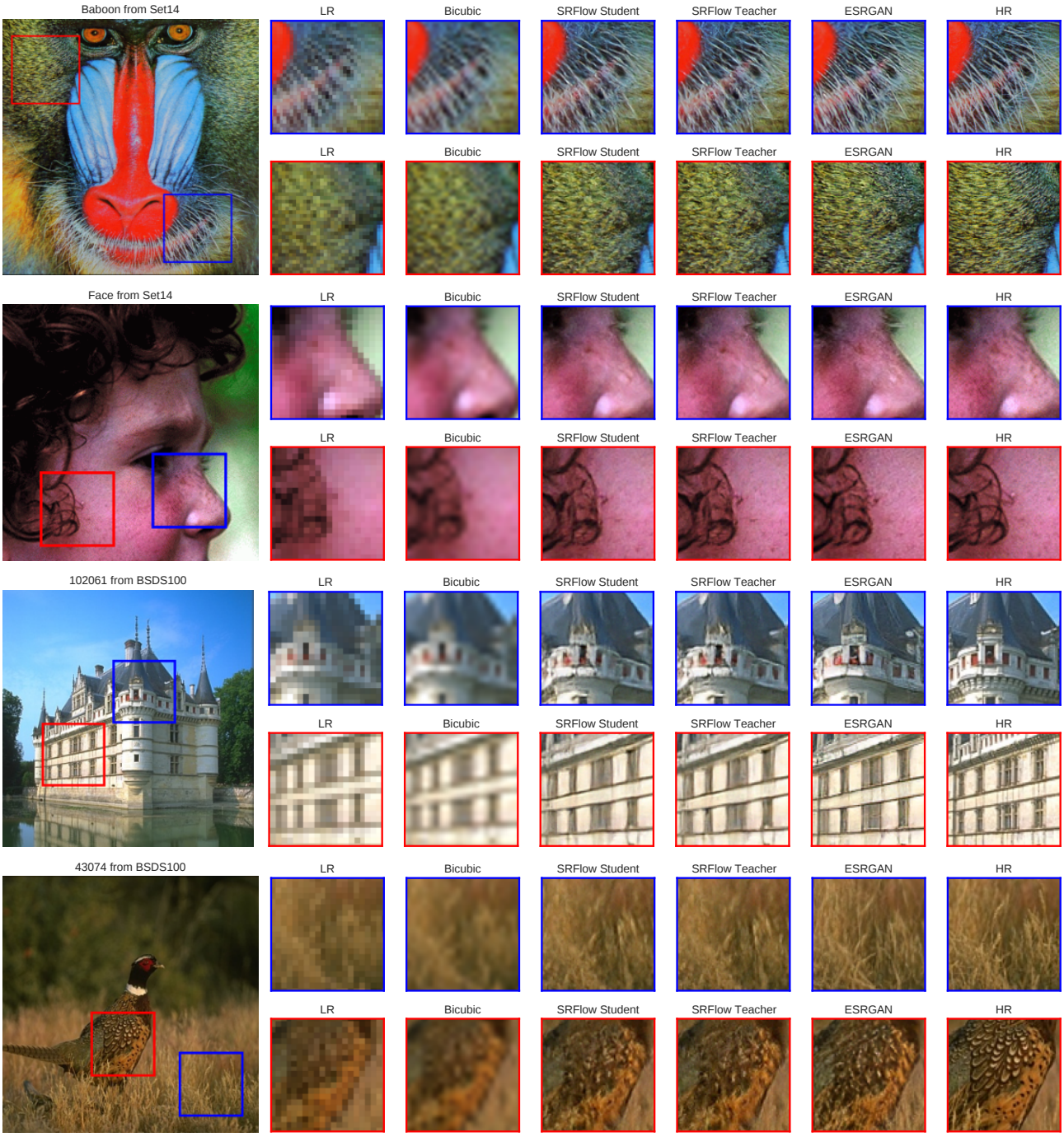

Figure 4. Qualitative results of SRFlow teacher and student for $\times 4$ scaling factor. The student model produces SR samples similar to the teacher ones.

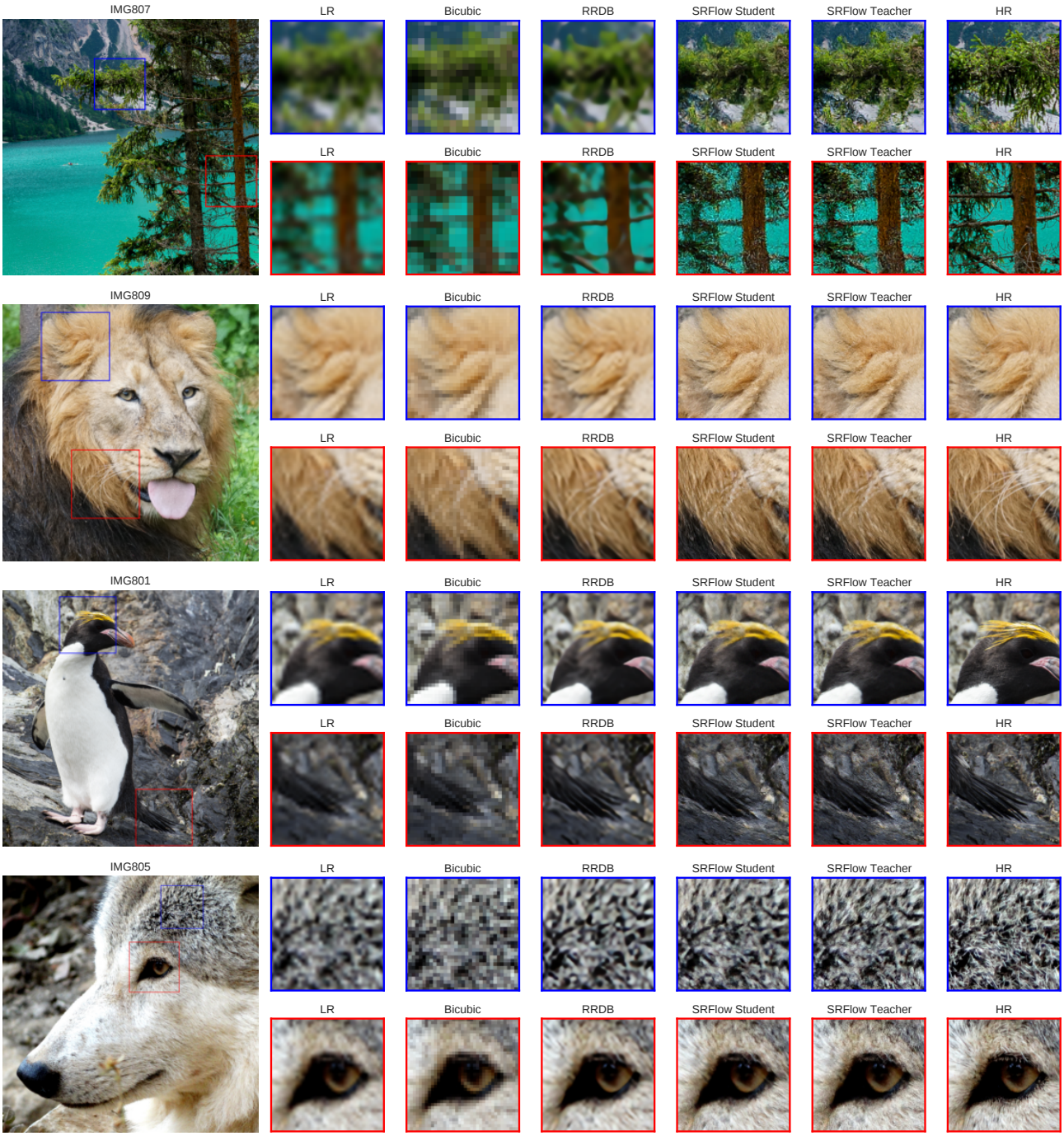

Figure 5. Qualitative results of SRFlow teacher and student on the DIV2K dataset for $\times 8$ scaling factor. On most samples, the student model demonstrates performance close to the teacher.