



iApp Image Upscaler: Perceptual Quality at 2× and 4×

Service version 1.0.0-20260907 · Measured 28 October 2026 · iApp Technology Co., Ltd., Bangkok, Thailand

Executive summary. We reduced 200 high-quality reference images from Wikimedia Commons to a quarter and a half of their size, enlarged them back with the iApp Image Upscaler, and measured how close the result is to the original with LPIPS, a perceptual distance that follows human judgement. At 4× the upscaler's output was closer to the original than plain bicubic enlargement for 196 of 200 images, with a mean LPIPS of 0.221 against 0.403 (45% lower; difference -0.182 , 95% CI $[-0.196, -0.169]$). From JPEG-compressed inputs, typical of images on the web, the gap is wider: 0.251 against 0.531. As expected of an upscaler that reconstructs texture instead of smoothing it, its PSNR is lower than bicubic's (26.65 against 27.98 dB at 4×); PSNR rewards blur.

1 Introduction

The iApp Image Upscaler enlarges an image 2× or 4×, up to 16.8 megapixels, reconstructing edges, texture and lettering rather than blurring them. This paper measures how close its output comes to the true high-resolution image, on images it has not seen, against bicubic resizing, and states the limits of the evidence in Section 7.

2 Test Data

The references are 200 featured pictures from Wikimedia Commons, images the Commons community has reviewed for technical quality: photographs of landscapes, architecture, animals, plants and objects, and scans of paintings, prints and maps. They were drawn at random (seed 1337) from all featured pictures that are JPEG files at least 2,048 pixels wide, with an aspect ratio up to 2:1, under CC0, public domain, CC BY or CC BY-SA; author and licence are recorded for each. Each was reduced with a Lanczos filter to 1,024 pixels on its longer side, which suppresses sensor noise and compression, and serves as the reference. The images are not drawn from the collections the service's model was trained on; some featured pictures are published elsewhere too, so a chance overlap cannot be excluded.

3 Method

For each reference and each scale $s \in \{2, 4\}$ we made a low-resolution input by reducing the reference s times with an antialiased bicubic filter, under two conditions: *clean*, sent as PNG, and *JPEG 75*, saved as JPEG at quality 75 before sending, as most images on the web are. The service enlarged each input by s with its default photo model and returned PNG. Plain bicubic enlargement of the same input is the baseline. Both are compared with the reference:

- **LPIPS** [1]: a learned perceptual distance (AlexNet features, version 0.1) that agrees with human similarity judgements; lower is better.
- **PSNR** and **SSIM**: pixel and local-structure fidelity on the luma channel, excluding an s -pixel border, as is usual for super-resolution; higher is better.

Why PSNR alone misleads. A low-resolution image is consistent with many high-resolution images. Their pixel-wise average — a blur — minimises the expected squared error, so the smoothest answer scores the highest PSNR. An upscaler that draws plausible fine texture places it a pixel or two from the true texture and loses PSNR, although a person sees a sharper, more faithful image [2]. We therefore lead with LPIPS and publish PSNR and SSIM beside it.

Differences are paired by image; confidence intervals are seeded percentile bootstraps (2,000 resamples, seed 1337). The service handled all 800 requests; 0 failed.

4 Results

Table 1: LPIPS, lower is better. Δ is the upscaler minus bicubic, with its 95% CI; the last column counts the images where the upscaler is closer to the reference.

Scale, input	Images	Bicubic	Upscaler	Δ [95% CI]	Upscaler closer
4×, clean	200	0.403	0.221	−0.182 [−0.196, −0.169]	196 of 200
4×, JPEG 75	200	0.531	0.251	−0.280 [−0.296, −0.266]	200 of 200
2×, clean	200	0.196	0.148	−0.049 [−0.057, −0.042]	162 of 200
2×, JPEG 75	200	0.306	0.167	−0.139 [−0.149, −0.131]	197 of 200

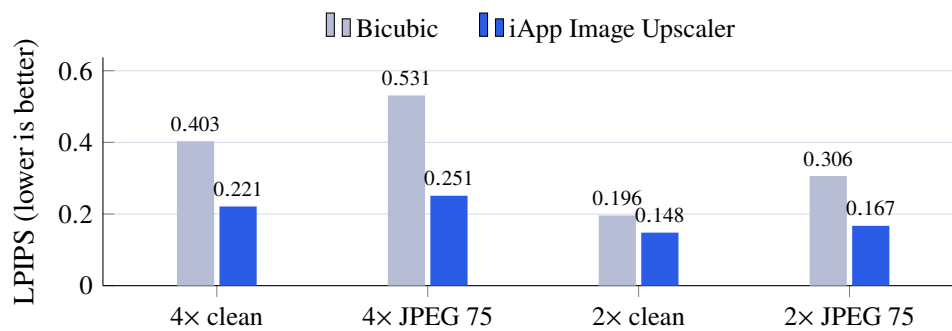


Figure 1: Mean LPIPS to the reference, 200 images per condition.

Table 2: Pixel fidelity on the luma channel: PSNR (dB) and SSIM, higher is better. The last column counts the images where the upscaler has the higher PSNR.

Scale, input	PSNR (dB)		SSIM		Higher PSNR
	Bicubic	Upscaler	Bicubic	Upscaler	
4×, clean	27.98	26.65	0.752	0.737	17 of 200
4×, JPEG 75	27.40	25.94	0.723	0.705	13 of 200
2×, clean	31.35	29.29	0.876	0.829	1 of 200
2×, JPEG 75	30.34	28.82	0.837	0.811	4 of 200



Figure 2: 4× from a clean input, 256 × 256 crops at full size. Left to right: bicubic, iApp Image Upscaler, reference. The crop is the most detailed window of each image, chosen automatically. row 1: *Worcester Guildhall Assembly Room*, Michael D Beckwith, CC0, Wikimedia Commons (LPIPS bicubic 0.565, upscaler 0.290); row 2: *Mangostim*, ArionStar, CC0, Wikimedia Commons (LPIPS bicubic 0.451, upscaler 0.302).

5 Discussion

Perceptually, the upscaler is closer to the original at both scales. The LPIPS gain holds in every condition, and its confidence interval excludes zero in each (Table 1). The gain is largest at 4×, where bicubic has the least detail to work with, and smallest at 2× from clean inputs, where bicubic is already close and remains the closer of the two for 38 of 200 images.

Compressed inputs gain the most. From JPEG inputs, bicubic enlarges the compression artefacts along with the image, and the upscaler’s lead widens at both scales (0.251 against 0.531 at 4×, closer on 200 of 200). For images taken from the web or from messaging applications, this is the common case.

Lower PSNR is the expected trade. The upscaler’s PSNR is below bicubic’s in every condition, and SSIM follows the same direction (Table 2). This is the signature of reconstructed texture described in the Method section, not a loss of fidelity a viewer would see; the example figure shows the difference.

What the output is not. Fine detail absent from the input is inferred, not recovered: in Figure 2 the finest ornament of the ceiling is simplified, not restored. Small text or a distant face comes out sharp but not necessarily true, so an enlarged image is not evidence of what a low-resolution source showed.

6 Production Performance

Time per image. Fixed input sizes were sent one at a time to the production service; it reports the time spent enlarging (model) and the total including decoding and JPEG encoding.

At the largest 4× output, one request every 1.25 s is about 2,887 images an hour; requests beyond that queue. PNG output takes longer to encode than JPEG at the same size.

Public API. A seeded sample of 40 of the 4× clean inputs (seed 1337) was sent through <https://api.iapp.co.th/v3/store/image-generation/image-upscaler/upscale> from Bangkok on 28 October 2026, one at a time. All 40 succeeded and 40 returned pixels identical to the production service’s output, so the quality figures above are what a customer receives. End-to-end time, including upload and the PNG download, was 0.51 s at the median (90th percentile 0.68 s, 95th 0.69 s) for 1,024-pixel outputs.

Table 3: Service time, seconds, median of 5 runs, JPEG output. The fast model, offered for throughput, is 4× only.

Input	Scale	Output	Model	Total	Fast model
256 × 256	4×	1,024 × 1,024	0.06	0.06	0.02
512 × 512	4×	2,048 × 2,048	0.26	0.28	0.05
1,024 × 1,024	4×	4,096 × 4,096	1.16	1.25	0.24
960 × 540	4×	3,840 × 2,160	0.57	0.61	0.11
1,024 × 1,024	2×	2,048 × 2,048	0.23	0.24	–
1,920 × 1,080	2×	3,840 × 2,160	0.50	0.55	–
2,048 × 2,048	2×	4,096 × 4,096	0.97	1.06	–

7 Limitations

- **Synthetic reduction.** The inputs were made by reducing a clean reference. Real low-resolution images also carry camera blur, noise and repeated compression; the JPEG 75 condition covers only the last.
- **Reference size.** References are 1,024 pixels on the longer side, so 4× inputs are 256 pixels; the service accepts inputs up to 1,024 pixels at 4×, whose quality we did not measure against a reference.
- **One model.** Only the default photo model was scored for quality. The illustration and fast models were not.
- **LPIPS is a proxy.** It agrees with human judgement better than PSNR or SSIM but is not a viewer study.

8 Reproducibility

`get_data.py` rebuilds the reference set from Wikimedia Commons with the seed above; `bench.py` makes the inputs, calls the service and scores each output; `gen_results.py` and `make_figure.py` produce every figure in this paper. The image list with authors and licences, the per-image scores and the scripts are available from iApp Technology on request: sale@iapp.co.th.

References

- [1] R. Zhang, P. Isola, A. A. Efros, E. Shechtman and O. Wang. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. *CVPR*, 2018.
- [2] Y. Blau and T. Michaeli. The Perception-Distortion Tradeoff. *CVPR*, 2018.

About iApp Technology. iApp Technology Co., Ltd. is a Thai artificial-intelligence company providing OCR, document understanding, speech, identity verification and large-language-model services for the Thai language. Web: iapp.co.th · E-mail: sale@iapp.co.th · Tel: +66 2 124 4041