

Vyb Fidelity + Editability Benchmarks

Note: we will be publicly releasing the benchmark on Hugging Face as part of launch

The Fidelity and Editability Benchmarks measure how accurately AI models reproduce typefaces and colors and how effectively they perform non-destructive edits. It covers 145 fonts, 100 Colors, and 200 Edits. For fidelity outputs (fonts and colors), every output is scored against the exact reference and for editability, every change is verified against an independent LLM reviewer.

The rankings below show how each of the models perform across each of these dimensions:

Model	Typography	Color	Non-destructive Editing
Vyb 1.0	1	1	1
ChatGPT Image (gpt-image-2.5)	2	2	3
Muse (muse-image)	3	3	2
Recraft (recraft v4.1)	4	4	4

Results

The following are the detailed results for each category.

Typography

We picked 145 [Google fonts](#) for this benchmark. Each model is asked to produce the word *The* in each of the fonts. We use these fonts because they are publicly available and free-to-use. As a result they are commonly seen online, which means each of the models should be familiar with them.

All the models other than Vyb output pixels. So we compute the error as follows:

First, we generate the same word, in the same font programmatically, using the Python Image Library ([PIL](#)). This is our ground truth. Then we take the generation of each model and crop and scale it so that it maximally aligns with the programmatic generation. Finally we compute the Intersection over Union. This is the fraction of pixels that are part of the text in both outputs as compared to the ones that are in at least one.

Ideally perfect font generation would result in a score of 100%. However, because we are comparing pixels that come from generations with limited resolutions, have to deal with anti-aliasing, and imperfect scaling / positioning algorithms, this measurement is inherently noisy.

To quantify the magnitude of the noise, we compared programmatic font generations with themselves when generated at different scales and positions. For these perfect generations, our metric produced a mean score of 95.8%. Therefore, any mean score approaching this value should be treated as a perfect generation.

In the table below, we report the Intersection of Union across the 145 fonts. Each model is run 3 times per font, and we average every font across the three runs.

Model	Mean	Median	20th %ile	10th %ile	5th %ile
Vyb (ours)	95.3%	96.0%	93.7%	92.5%	90.8%
ChatGPT Image (gpt-image-2.5)	78.2%	86.7%	70.1%	44.7%	29.4%
Muse (muse-image)	59.8%	66.2%	41.8%	29.5%	22.1%
Recraft (recraft v4.1)	51.2%	53.0%	32.1%	23.8%	14.2%
<i>measurement noise floor</i>	95.8%	96.0%	94.0%	93.1%	92.2%

As we can see here, Vyb stays very close to the measurement noise floor. Perhaps most striking is the dropoff. While in the median ChatGPT is able to achieve a score of 86.7% as compared to Vyb's score of 93.7%, in the tail ChatGPT drops to a score of 29.4% as opposed to Vyb's, which maintains a score of 90.8%.

Color

To assess how well models can follow color specifications, we pick 100 colors. For each color, we have each model generate a solid rectangle in that color. Then we compute the mathematical difference between the pixel color that the model produces and the specified color. Our pixels are encoded in three channels (Red, Green, and Blue). So we take the absolute difference for each channel (out of 255) and report the max across those channels. We do this three times for each color and model. The following table reports the results:

Model	Mean	Median	80th %ile	90th %ile	95th %ile
Vyb (ours)	0.00	0.00	0.00	0.00	0.00
ChatGPT Image (gpt-image-2.5)	4.12	3.83	5.07	5.70	7.33
Muse (muse-image)	5.63	5.67	7.07	9.03	10.35
Recraft (recraft v4.1)	27.79	26.00	39.37	47.40	49.50

This benchmark shows that ChatGPT and Muse are able to get close to, but not reproduce the exact color. Recraft has a more challenging time matching the requested color. Vyb, on the other hand, reproduces the color exactly.

Non-destructive Editing

We define non-destructive edits as those where the pixels of the regions of the image that are not being edited do not change. We measure this through the following approach. We prompt each of the models with a starter prompt so that they generate a similar starting visual. We then ask each model to perform the same edit. Our edits come from the following categories:

Category	What it changes
color	recolor a text element or shape
text	change the wording of text
add_asset	add a provided asset (logo / badge / icon)
image_replacement	swap a photo/graphic for a provided one
combined	two or more changes in one instruction

Category	What it changes
decor_addition	add a decorative element (line art, doodles)
removal	remove an element
font_size	enlarge or shrink text
logo_replacement	replace a text/logo with a provided logo
position	move, re-crop, or re-scale an element
decor_replacement	swap a decorative element for another
group_formatting	align / evenly space a group of elements

We confirm that the models have performed the edits through an LLM judge and record the total number of pixels changed. We perform this measurement across 200 edits, taking the average over three runs. The following table shows the results:

Model	Mean	Median	80th %ile	90th %ile	95th %ile
Vyb (ours)	11.3%	6.6%	19.4%	31.5%	36.9%
ChatGPT Image (gpt-image-2.5)	95.9%	97.6%	98.9%	99.3%	99.5%
Muse (muse-image)	90.7%	95.6%	97.9%	98.5%	98.9%
Recraft (recraft v4.1)	99.2%	100.0%	100.0%	100.0%	100.0%

All models, other than Vyb, change the majority of pixels most of the time. We dug into the data to understand why Vyb sometimes changes a larger fraction of pixels. This most often happens due to larger-scale edits (e.g. adding an asset or replacing an image) that warrant the substantial change in the number of pixels.

Benchmark Rationale

There are many cases where strict fidelity / specification is critical to a design. Users may request a specific font, color, or asset be used. Changes to this are often unacceptable to the user. We use mathematical metrics to assess these changes because they may simultaneously be hard-to-detect by a casual reviewer, yet critically important to the owner of the asset.