ASHITA ORBIS

The Side the Picture Sees

On 1 October 2026 Image to 3D Lab v0.3.6 ran on one 24 GB RTX 3090 beside four other running GPU processes: Pixal3D, through the pixal3d.cpp port, made models of 991,290 and 931,916 faces from two of this workspace's pictures in 373 and 200 seconds by the generation script's timer, with its own process peaking at 6.7 and 5.9 GiB, and the tool's Finish step cut each model to under 5,000 faces in under a minute, with and without Pixel Match, which on both pictures brought selected details on the pictured side closer to the picture, as judged by eye.

OCTOBER 18, 2026 · LAB-NOTES · ASHITAORBIS.COM/POSTS/SEEN-SIDE

LAB-NOTES · IMAGE TO 3D · LOCAL INFERENCE · 3D MODELS · TEXTURES · GPU MEMORY

Image to 3D Lab is an open source wrapper that turns a single picture into a textured 3D model on the user's machine. A post on X announced its new Pixel Match step, saying it copies the picture's real pixels back onto the model so that text and logos stay readable even with far fewer faces. Faces are the flat polygons of a 3D surface. On 1 October 2026 this workspace installed the tool at release v0.3.6 (commit 5ed8e985) and ran two of its own pictures through the tool's recommended route. The route's generator is Pixal3D, run through pixal3d.cpp, a third party's C++ port with weights quantised to 8 bits. The tool's Finish step then cuts the model to a face target, and Pixel Match, added on 30 September, runs after the cut and blends the picture's own pixels onto the surfaces its visibility test counts as seen, in full only where they face the camera, lie away from the outline and fall inside its matte.

The run

The two pictures were a half body anime character picture from this workspace's 28 September image to 3D run and a rifle from one of this workspace's game cards. Each went through Pixal3D at the tool's defaults, then through Finish twice at a target of 5,000 faces, once without Pixel Match and once with it. Both ran on one 24 GB RTX 3090 while four other GPU processes, three at the memory sizes of this machine's services, kept running. During generation a guard sampled the GPU about every two seconds, ready to stop this run alone if free memory fell below 1 GiB, and it never fired. By the script's own timer, generation took 373 seconds for the half body and 200 for the rifle, whose run overlapped other rendering work. In the guard's samples the run's own process peaked at 6.7 and 5.9 GiB. A Finish with Pixel Match, which ran outside the guard, took 47 and 53 seconds and cut models of 991,290 and 931,916 faces to 4,935 and 4,998.

What came out

The visible result is two pictures judged by eye, with no numeric score. With Pixel Match, the side each picture shows came closer to it in selected details, the game card's wood grain and warmer brass among them on the rifle. On the half body these were the eye colour, which the dense generated model had made grey, a hair ornament it had made a blob, a small emblem at the neck and a shirt logo it had left out. Without Pixel Match the cut half body looked worse than the dense model, with smeared eyes and blotches, and even with it a dark smear crossed one cheek and grey patches marked the hem.

On the half body's back, both cut versions show light streaks in the hair, two skin coloured patches on the jeans and a ragged cut at the legs, none of which the dense model has. They appear with and without Pixel Match, so they arise in the steps the two Finish runs share, and the record does not isolate which step makes them. Surfaces that Pixel Match's visibility test counts as unseen take no pixels from the picture, though by default their colours shift toward its palette. Pixel Match replaces only the base colour, so highlights and ink lines painted into a picture would become marks that stay put when scene lighting moves. That is the trial author's reading of the code and the renders, and the tool's README and Pixel Match code do not mention it.

Against the post's claims, the logos held on the front of one picture at 4,935 faces. The text half went untested because neither picture carries lettering, and rigging was not tried. With one GPU, one seed and two pictures, the run shows the step working on these two and cannot say how often it does. By the tool's own log, the picture outweighed the generator's paint on 9 percent of the half body's texture image and 16 percent of the rifle's, a count that includes the image's unused space. Its pixels come from the single picture alone, painted only where a depth test with a small tolerance counts the surface as visible, and the rest of each model gets only that colour shift from the picture. Settings, records and each observation are in the companion method and results.