



ApPEARS

APPEARANCE PRINTING
European Advanced Research School



Fast impression painting using multi-color fused deposition modeling with a diamond hotend

R. Tonello¹, F.Abedini², D. B. Pedersen¹ and J. R. Frisvad¹
¹Technical University of Denmark. ²Linköping University
Email: rict@dtu.dk, jerf@dtu.dk

Abstract

Fused deposition modeling (FDM) is an easily accessible technology for additive manufacturing. The use of a diamond hot-end [FIG. 5] is one way to get multiple colors in FDM. As multicolour system, it has some disadvantages, such as the need to use a purge tower to exhaust every color before moving to the next one and the long printing times.

Our goal is to avoid the purge tower in order to improve speed and reduce waste in this type of multi-color FDM printing. This requires a different workflow. Our idea is to control the number of steps before switching to the next color and the length and time of the retraction of the filament when foreseeing movement without extrusion. However, increasing the speed worsens the accuracy and viceversa, so we focus on fast printing of impression paintings, where the accuracy is not an essential factor.

In addition, we want to corroborate these strategies with a supportive visualization tool that can process images, reorganize them according to the available color gamut of the printer and detect the number of points that we need on the buildplate for a given print job. We test our multi-color method by printing an impression painting corresponding to a given input image

Color threshold based segmentation of a given image



Color threshold based segmentation with superpixels in order to remove spare pixels (Fig.1)

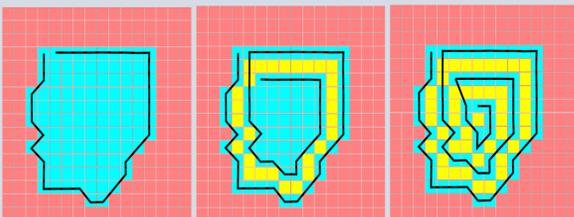
In the first part, we took advantage of superpixels and clustering techniques. We restricted the printer's gamut to six colors and we employed k-means clustering to partition the image into six clusters (i.e., $k=6$). We used superpixels to avoid the rigid structure of the pixels and detached single pixels in clusters.

Superpixel algorithms group pixels into regions with similar perceptual properties that adhere well to the image edges. Thus, we employed the simple linear iterative clustering (SLIC) algorithm to group pixels with similar values in the CIELAB color space into superpixels.

The use of superpixels improves the segmentation performance by reducing its complexity. For instance, if our target image (FIG. 1, left) is of resolution 250x187, SLIC can reduce the number of picture elements from 46750 pixels to 2028 superpixels.

In addition, superpixels provide more information than regular pixels. So, to further improve the segmentation, we supplemented the segmentation algorithm with the mean value of each superpixel. As a result, the k-means clustering algorithm prefers grouping superpixels with similar mean CIELAB values into the same cluster. FIG. 1 (right) shows the result of using superpixels and k-means clustering techniques to segment the image into 6 clusters.

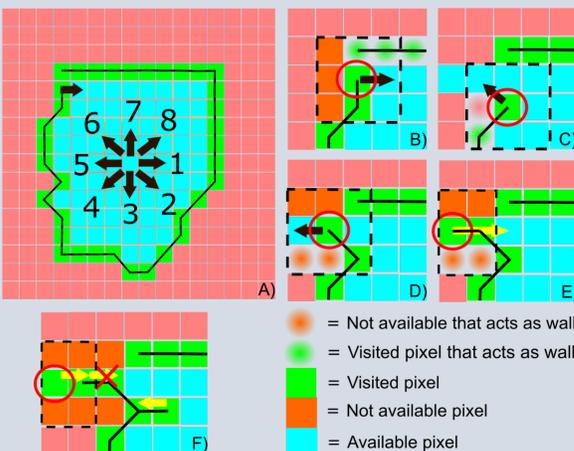
Creation of the G-code



Toolpath generation with contour detection. (Fig.2)

We can then abandon the use of purge towers and reduce the presence of artifacts and ooze through a smaller stepwise transition between colors during printing. However, these two improvements are inversely proportional, since the absence of an exhaust tower and the faster printing are sources for artifacts and decrease the overall accuracy.

To find a good tradeoff, we developed a customized G-code following two different approaches. Obtaining masks for each color given by the segmentation of the processed image, the first method consisted of detecting and storing all the points of the borders (light blue pixels in FIG. 2). After this, we excluded all the identified border points and repeated the previous step, with the only difference that we did not store the points this time (yellow pixels in FIG.2). Alternating a border with stored points and one without stored points, it was possible to create a concentric toolpath [FIG. 2].



Strategies adopted to define a concentric toolpath. (Fig.3)

When selecting extrusion direction, we try directions $v-1, v, v+1$ (with this priority). If the pixels in these three directions are all unavailable, we shift our preferred direction 90° clockwise, $v:=(v+1 \text{ mod } 8)+1$ [FIG. 3B], and we continue.

- If we reach a dead-end [FIG. 3E], the tool-head goes back a few steps without extruding and selects a new direction as before.
- If we move back for a maximum number of steps without finding a new direction, the toolpath breaks, and we continue from the next unvisited point in the image [FIG. 3F].
- Selecting a 5x5 grid, additional features are added to prevent that the toolpath breaks [FIG. 3B-C-D-E].

After producing the segmented image, we created an operational dataset of filament mixing ratios sorted by the minimum color space distance between the colors in the printer gamut. This sorting enables us to minimize the color transition between different hues during a print, which is otherwise the constraint of multi-color FDM that leads to use of purge towers.

Another approach is to define two filter grids, respectively 3x3 and 5x5 in order to have an idea of the neighbourhood of the detected pixel. This method enables a higher freedom to operate and in this way, it has been possible to define strategies that avoid singularities such as non-Euclidean geometries, which may incur overlapping of printed filament or too narrow print areas.

For this approach, we devised the following G-code generation strategy [FIG. 3]:

- Use of a preferred print direction (v) with clockwise ordering (numbers in FIG. 3A). We have a preferred direction along a coordinate axis ($v \in \{1,3,5,7\}$).

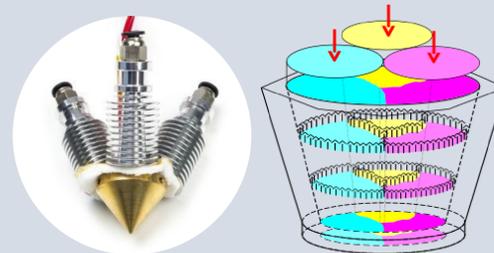
Visualization tool



Visualization tool (right image) with points of extrusion (Fig.4)

After designing the G-code tool path, we implemented a supportive visualization tool (FIG. 4, middle) in order to understand if all points were correctly transcribed and the fastest transition between colors was respected. The black spots in FIG. 4 (middle) are excluded points. These are points without extrusion or print areas too small to be meaningfully printed. We inspected the outputs of our different approaches using a commercial G-code viewer. An example is in FIG. 4 (right).

Experimental part and conclusions



Nozzle tip diamond hot-end and cross section (Fig.5)

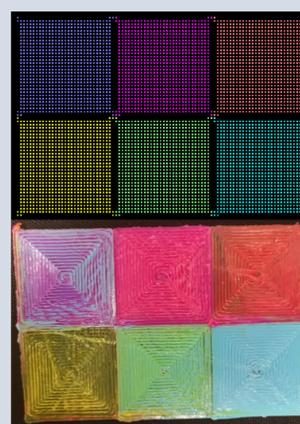
After creating the G-code file, we tested it on a diamond hot-end 3D printer. Diamond hot-end is a FDM technology, where multiple filaments coalesce in one hot end [FIG. 5].

With the aim of reducing the ooze and creating some fast prints without a purge tower, we calibrated the printer accordingly. To reduce the oozing effect, we predicted the amount of visible material dropped and dragged by the extruder across the buildplate after stopping the printing and computed the number of points in advance to start retracting the filament. We used trial and error until we reached a promising result for the square with concentric infill [FIG 6].

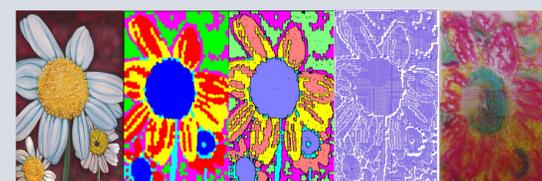
We used a very similar approach when switching colors. By computing the length of filament needed for the color transition, we inserted the switch to another color a number of points before completing the section, so as to reduce the color transition to the center and the border of each printing area. We tackled the experiments by designing a simple version of a painting: a set of squares next to each other to prove the concept [FIG. 6].

The extruder prints at 220°C with 15 points of retraction length for each no extrusion movement, while it counts 15 points ahead to switch to Cyan, Magenta and Yellow, but 35 points for the combinations of them, since higher pressures are needed to effectively extrude 2 filaments at the same time.

We printed three layers in our tests because the result with one layer became too brittle to be detached from the build-plate without breaking. Even though some simple geometries can be printed without the use of the purge tower and with few precautions in terms of retraction length, more complex geometries are still hard to print without some additional aids [FIG. 7]. However, it is possible to print fast impressions as shown in the figure below and we are confident that it is ultimately possible to improve the accuracy.



G-code points of six squares in the visualization tool (top). Six 3D printed squares without purge tower (bottom) (Fig.6)



Workflow from the image to the 3D printed part (Fig.7)

References

Fig.5 (left): Nozzle Diamond Hot End (RepRap World) -<https://reprapworld.dk/extruder/hot-end/multi-extrusion/diamond-hotend/>



ApPEARS is funded by the European Union's Horizon 2020 programme under the Marie Skłodowska-Curie grant agreement No 814158