Encoding Animations of Image Segments using Barcodes

Background

One use for QR codes and other barcodes is to provide instructions: a URL for a video containing instructions on how to use some device (for example, a vacuum cleaner) could be encoded, printed as a sticker, and placed on the device. However, this process requires that the instruction video be available on a server, and that the user is connected to the internet (or has sufficient bandwidth to stream a video). Qingyang Chen worked on a project for CPSC 490 last semester which attempted to solve this problem by encoding animations into 2D barcodes. Using said system, a “creator” would first take a picture of the object or scene that they wished to animate. They would then specify polygons on the image and the transformations of those polygons necessary to produce their animation. Finally, these values could be encoded into a QR code. The “viewer” would then be able to generate a video of the specified animation by taking a picture of the device, including the barcode.

Qingyang’s system allowed for the specification of polygons with a limited number of vertices as well as their transformations. The system for matching the encoded polygons to the shapes seen in the image first used a Canny edge detection filter, followed by the Hough Line segment extraction algorithm to find line segments contained in the image. This was particularly effective when the specified polygons matched distinctive edges in the image. The system then determined the closest match for each line segment of the specified polygons. As each polygon was animated, it left behind a “hole” in the image, which was filled using background implantation.

Proposal

My goal is to improve upon this system by using an image segmentation algorithm rather than manually specified polygons. Under my modified methodology, the creator will first take a picture of the object to be animated, and an image segmentation algorithm will be

---

1 http://zoo.cs.yale.edu/classes/cs490/16-17b/chen.qingyang.qc43/
applied, dividing the image into its distinctive parts. Next, the segments must be enumerated in some deterministic way. The transformations to be applied to each segment can then be encoded as a 2D barcode. The viewer should then be able to take a picture of the object and the barcode, and subsequently view a video of the specified animations applied to the image.

I expect that generating videos in this way will solve some of the problems of the previous approach; in particular, this approach will support unusually shaped objects and parts, and may not have the same issues with parts of the object being “left behind” when transformed. Additionally, it may make the creator’s experience simpler and more intuitive, as they will not have to specify vertices but can instead drag and drop the available shapes.

Challenges

The most central problem is the task of enumerating the segments and encoding their transformations using the ~194 bytes available in the typical QR code. This is a difficult and important problem because the viewer may not take their photo from the same position as the creator, and so their photo may have additional (or fewer) segments. If the segments are enumerated or encoded in a non-optimal way, the presence of an additional segment could create an “off-by-one” error for all of the other segments. A solution might enumerate based on distance from the sticker, or encode an approximate position for each segment into the barcode; however, I will likely need to try a number of techniques and heuristics for encoding and matching segments in order to determine what works best.

Using segmentation also brings the system much closer to true augmented reality. A modified “true AR” system might allow the viewer to view the specified animation on top of the live feed from a camera, rather than generating a video based on a single image. If the system already supports generating animations when the distance to the sticker and angle to the sticker are modified slightly from the creator’s image, then the main challenge is a computational one: the segmentation algorithm and subsequent image processing must be run on each frame. Note that such a system would only work within a limited range of angles to the barcode and distances from the barcode; if the viewer moved too far from their initial position, the specified 2D animations would no longer make sense.

Deliverables

I will develop a Creator interface which takes an image, applies a segmentation algorithm, and allows the user (creator) to specify various transformations for said segments
or groups of segments. I will also develop a Viewer program which takes a photo and barcode data, and produces a video of the specified animation. I will not be working on the actual generation or decoding of a QR code, nor will I be developing new algorithms for image segmentation.

As a stretch goal, I will extend the viewer program to take a sequence of “live” images and generate a video of the specified animation on top of the frames of the sequence.