Ignore. Import content3 from here.

use author=string

2018-05-27

The Geometric Motivation:

This is a short section/chapter covering the motivation to use Geometric shapes for our segmentation, and some mathematically nice qualities of these shapes.

Platonic Solids:

From the ancient times there has been a special place for what was known then as Platonic solids. These were thought to be sacred shapes that described something True, and for our intents and purposes that might as well be true.

These shapes are indeed special, including but not limited to the cube, the pyramid, and the dodocahedron.

Essential qualities of these are that they are polygons, and therefore also convex shapes. As a curiosity these were traditionally combined into a single shape, called the Metatron's Cube.

Convexity:

A set, C, is said to be convex if $\forall x, y \in C$ and $t \in [0, 1]$ the following is always true:

$$(1-t)x + yt \in C \tag{1}$$

This is trivially true, perhaps, for polygons, but it is essential when considering our project.

Projection of Convex Shapes:

Basic Assumption: The projection of a 3D, Convex shape to a 2D image-surface will form a 2D Convex Shape. (This assumption is true, but will not be proven here)

Knowing that this is the case, the following preposition is an important bit of logic for our project.

Geometric assumptions, and our Project:

If we assume that a body is a combination of polygons, and other convex shapes such as spheres, then from an image of a set of polygons we can infer a 3D shape.

We do this every day when we assume that, e.g. an apple has a 3D shape, even though we can only see an image of it.

When we look at a head, we will assume it has a spheroid upper portion, and a jaw-structure which is similar to a bumpy semi-cylinder.

The image we see is undeniably 2D, and yet we have an underlying assumption of the shape based on knowing key-facts, and associating them with keypoints.

Conceptually then; it makes sense to endow the machine's vision with these same suppositions. Essentially to couple keypoints with geometric shapes. Using pose to extract rotation and keypoints then, we can theoretically form a human being composed entirely of convex shapes, and as noted this can be extended into assumptions as to the 3 dimensional shape of things.

This is not unlike how people have created human figures in video-games.

Vision and Segmentation Geometrically:

Vision in human beings is not an easy thing to emulate. Specifically our vision seems to be filtered through visual centers, creatively named V1, V2, etc.. What these visual centers seem to do is isolate qualities about the image you're perceiving. `https://en.wikipedia.org/wiki/Visual_cortex`

What we may call the secondary portion of vision relates to the projected expectations we have. That is, if a face turns to the side, we expect the head to expand backwards relative to the face, and we would be surprised if it did not. (See `https://en.wikipedia.org/wiki/Mental_rotation`)

In terms of what is happening in our human vision, there are theories to support we are operating on what's known as the Dorsal Stream. (`https://en.wikipedia.org/wiki/Two-streams_hypothesis#Dorsal_stream`)

In other words; in our human vision there are expected results from rotation of an object that stem from our experience in a three dimensional world. Borrowing some of this in an attempt to improve upon the machine vision, seems to us to make sense.

This is part of what Hinton (et. al.) uses to create the theory of Capsule Networks. (For more on CapsNet, we suggest this series of articles: `https://medium.com/ai%C2%B3-theory-practice-business/understanding-hintons-capsule-ne`

Our Concept:

For these reasons our algorithm was created to use geometric relations to generate a collected set of 2D convex shapes that change and morph with respect to the pose. The shape of our segmentation is a collection of ellipses and polygons, which with more time can be expanded to heuristically understand what possible 3D shapes form these on an image-surface.

For the 3D underlying shapes one has to actually rotate the object, remembering the previous rotations, keeping track of the object to ensure one does not "learn" the wrong things, and from this infer the correct geometric shape(s). (This would require us to create a LSTM or something similar, and was for that reason not included in the current scope.)

On Draw-functinos

We are implementing several different draw-functions for head, jawline, torso, arms and legs.

Head This function finds keypoints; eyes, nose, ears, and neck. Generating a semicircle or ellipse that covers the base structure of the back of the cranium.

Jaw	This draws a set of vertices that form a square on bottom, and vertices at about 45 deg angle from the eyes to cover noses or chinbones.
Torso	The torso is drawn in as two opposing semi-ellipses, which can be thought of as a slightly chubbied hour-glass.
Arms & legs	These are in our model assumed to be approximately cylindrical. So we draw thick lines or squares that cover these, keeping with the theory that the 2D projection of a cylinder, or cone-section (non parallel cylinder sides), will form a square-like shape.
Hands	Since our Pose-library doesn’t feature hands, we decided to simply cover these by semicircles that somewhat cover the range of motion that these are likely to be found in.

”gradient”-function:

There is also a somewhat peculiar version of a Laplacian manipulation in the top section of the draw-file. What this one does is taking an edge-image and manipulate it using the intermittent levels of the Laplace Pyramid.

Starting on the bottom, or a level above, it will dilate and erode, i.e. close, the particular level of the Laplace Pyramid. It then adds in the information from the next level of the Laplace before performing the PyrUp function.

The theory here is to use the various levels of the edge-image, in order to create something comparable to the filter generated from our geometric-shapes dummy.

It seems to work reasonably well,