**Method:**

Segmentation is a process to extract the counter of the shoes from the input images. It needs to be applied on the retailer shoes image and user image respectively, and the methodologies are different. The retailer shoes images from the brand’s official website usually have a pure color background. Since the content of the retailer shoes image is simple, one way to extract the counter of the shoe is treating image as function, and implement a range map operation which will extract all pixels or super pixels having different color from background. Compared with the retailer shoes image, the user image contains more complex information. To extract the shoes from the user image, the image needs to be represented as graph, and use the foreground/background segmentation with min-cut/max-flow graph cut algorithm. The foreground/background segmentation could create a binary mask where only the pixel or super pixels which are similar with the target are labeled. A general method to find those pixels is minimizing the energy function as shown below.

Energy Function:

I represents image, represents the key values of foreground or background, x is a binary function, f and b are functions which determine the weight of edges between pixels and target. Min-cut/max-flow graph cut is a globally optimal solution to solve such a minimization problem. By representing the images as graph, a flow network can be constructed for images. A flow network is a directed graph where each edge has a non-negative capacity, and the capacity is usually determined by the color similarity and proximity between pixels or super pixels. The Ford & Fulkerson algorithm, an algorithm that can compute max flow in a flow network, can sort pixels or super pixels of image to two parts where the sum of capacities in each part reach maximum. In this project, a pixel or super pixel from the shoe is selected as the target to do the foreground/background segmentation.

**Implementation:**

In the project, all super pixels were created using simple linear iterative clustering (SLIC) super pixel algorithm, which clusters pixels in the image plane based on their color similarity and proximity to generate a specific number of super pixels. In python code, the “skimage.segmentation.slic” function was used. The first two parameters of the function “n\_segments” and “compactness”, which influence the number and the shape of the super pixels, are determined after a series of tests while others keep the default values. Based on the collected data, it was assumed that all input images have a white background. To obtain the shoes mask from the retailer shoes image, a function which can detect all non-white pixels and change pixel values to black was implemented. A similar result could be obtained with a similar function which can detect all super pixels having non-white center pixel and change those super pixels to black. For the user image, because it is hard to find a global optimal function that determine weight of edges between super pixels, the foreground/background segmentation code we wrote didn’t perform well, so… (the actual code we used for foreground/background segmentation)

图片包含 游戏机

描述已自动生成

Figure 1. sample output

**Limitation & Issues:**

The current strategy for the segmentation of retailer shoes image is a rough sketch. There are few limitations needed to be imposed on the input image to ensure a reasonable output. For all input images, the color of the majority of the shoe cannot be the same with the background color; Otherwise, the final output will be incomplete. As mentioned before, the background color of the image must be pure. The program will fail with a colorful background image. Besides, the current methods also have some unsolved bugs. One thing is that there will be some shadows in the retailer shoes image as shown in figure 2. Since the code extract the shoe by detecting the non-white pixels or super pixels, those shadow parts will be kept in the final output. Another thing is that the current super pixels generating function could not generate a well output for all input images. For some images, the parameters of the function need to be edited to get a reasonable output.

鞋子的特写

描述已自动生成

Figure 2. shadows in the retailer shoes image

Reference:

<https://medium.com/@darshita1405/superpixels-and-slic-6b2d8a6e4f08>

Week7 lecture slide and capture