**PROGRESS REPORT**

**Instead of just finding some model or architecture and doing some hyper parameter tuning to complete the task, I chose to find the root cause of the specific problem and attack on that so that the problem arise from the task is of some actual use to the fynd team.**

**The problem of Image segmentation can be thought from a very different direction:**

**Use the present pix2pix model to generate a segmented image and then use GAN to produce the final image as we need. This way we can use some state of the artpix2pix models to work on our network and we just need to develop a GAN according to our needs.**

**Example: If hairs are not correct then we can make a GAN which stylize hairs getting a more profound segmented output easily.**

**Approach:**

**Find root cause -> Explore Literature (find inspiration) -> check for some pre trained model that can act as base -> Implement pretrained model or develop own model -> check performance -> repeat.**

**Problem in the background removal:**

Although pix2pix loss-based model works well but they lack in segmenting out minute details like hairs of the model or jewels etc.

The poor performance of the model can be attributed to the following reasons:

1. Input size(resolution) of the deep learning models is quite low like (256 X 256) by reducing high resolution model image (HD images) to such low resolution intrinsically remove much details as it gets lost in with the background pixel.
2. The dataset that we use also plays a very important role as model’s best performance is rarely better than the ground truth, thus ground truth images should also be perfect, more importantly they should have hairs (or the minute things that we want in the final output).

**Addressing problem one i.e. deep learning model**

1. We can slice the image into several images whose size is same as the ideal input size of the model so that minute details are not lost in down sampling the image to the model size.
   1. After removing out background from each image we can sew the image together.
2. There is roughness in the model output because the colors at the boundary of the are quite similar thus pix2pix loss fails to distinguish.
   1. My further exploration revealed that in majority of images there exists some difference among different channels (R, G and B). So, if we apply same model across different channels then the overall performance can be increased.
   2. If the colors are same across all the channels then we can apply cross masks like R channel of image and G channel of the mask.

**Addressing the problem of data set**

1. To have a good quality of dataset, have to automate the process of creating masks as the cost of designating a human to perform the task is very slow.
   1. So, to **generate** the mask, I got the inspiration from the word “generate” to use GAN networks develop masks.
   2. One more method that can be used is a certain combination of Auto Encoder and Decoder as in the literature also these are known to be performing well in generating some images are more consistent than GAN.

GAN generates masks and discriminator checks if the mask suits the need or not. Or adding a GAN in front of the Auto Encoder to generate better masks.