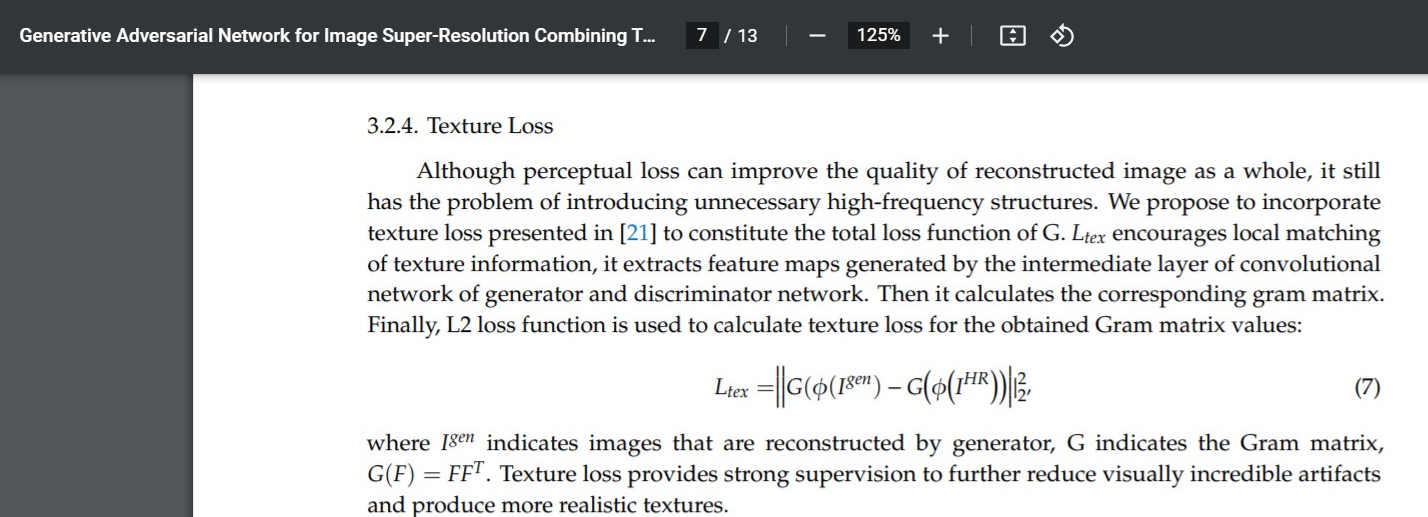
**Novel Architecture features:**

* DIV2K with data augmentation OR RealSR?
* GAN approach -- WGAN-GP loss
* Resnets / Densenets - 16/32 Resblocks
* No BN
* // Progressive Upsampling Super-resolution -- In fact, we can have progressive downscaling & then progressive upscaling module?
* End to end (HR->LR(with noise)->SR) network better ?
* Dataset: For training, we use the RGB input patches of size 96×96 from LR image with the corresponding HR patches & use augmentation.
* Loss components: Content(L1), adversarial(WGAN-GP), perceptual(VGG19) [perceptual loss covers: feature reconstruction + style reconstruction + L2 ]

**To resolve:**

* ? EDSR says: We pre-process all the images by subtracting the mean RGB value of the DIV2K dataset.
* New loss function
* Metrics - LPIPS, VIF, IFC, NIQE, Ma, BRISQUE,PSNR,SSIM,PIQE
* which color space is best for image super resolution? Frequency domain analysis ?
* minc loss -- ESRGAN?
* Texture loss?



**Random Thoughts:**

* Should we use the intermediate output of CAR/RealSR directly as the LR input for our architecture?
* Does it make more sense to use smaller Kernel size in the starting resblocks as it can abstract the smaller details //& larger ones in ending ones?

OR simply use all and concatenate

* what if we have >1 generators? Maybe 1 each for each of R, G & B...i.e.What if we try to super resolve each channel individually, & then try recombining ?
* In an end-to-end network, maybe calculate loss on LR & update the weights, next use LR->SR for updating weights in the next part.
* ELU/Leaky ReLU instead of ReLU?
* What if we used different color spaces to capture different aspects for SR & then, combine the results somehow(maybe conv layer concatenation just before upscaling).. Could help capture various advantages of each color space.. eg: robustness to lighting changes
* Densenets with Channel Attention? -- already been implemented, Resblocks perform better with CA.
* Why no Fully connected layers in any architecture? simple linear transformations might be helpful too.
* what if the pooling layers in RCAN CA are replaced by a Fully Connected layer?

**Extra Points to remember:**

* If we use VGG, use Alexnet weights coz LPIPS uses that
* visual patterns very high-dimensional highly correlated