	Cat			Dog			Otter		
n	RMSE1	RMSE2	RMSE3	RMSE1	RMSE2	RMSE3	RMSE1	RMSE2	RMSE3
2.0000	1.5653	22.412	37.4346	1.3603	30.392	49.9611	1.2632	20.002	35.7770
3.0000	3.2015	22.577	18.5688	2.8127	30.486	24.9647	2.5832	20.121	19.1056
4.0000	6.2896	23.185	10.1153	5.7467	30.868	13.1039	5.1778	20.586	10.7960
5.0000	12.515	25.463	6.5855	11.64	32.382	8.0754	10.231	22.301	7.5487

In the program we changed firstly, last n bits of each image. At this stage, if n is less, image will be less altered. For example if n is 2 a greyscale value of a pixel can increase or decrease maximumly 3, and if n is 3 this will be 7. Therefore, when n is less, rmse will also be less.

For second stage, less n bits means again less rmse but, this is not as first stage.

At the third stage, if we take more bits, when we place them as in the original image, we will see more similarity and less rmse values. This is because, if we kept 2 bits in first stage, the data we gathered is so limited. When carried bits are 5, the greyscale values rounded closely compared with 2 bits. As the result, if we kept more n bits, the final image will be more similar to original image.

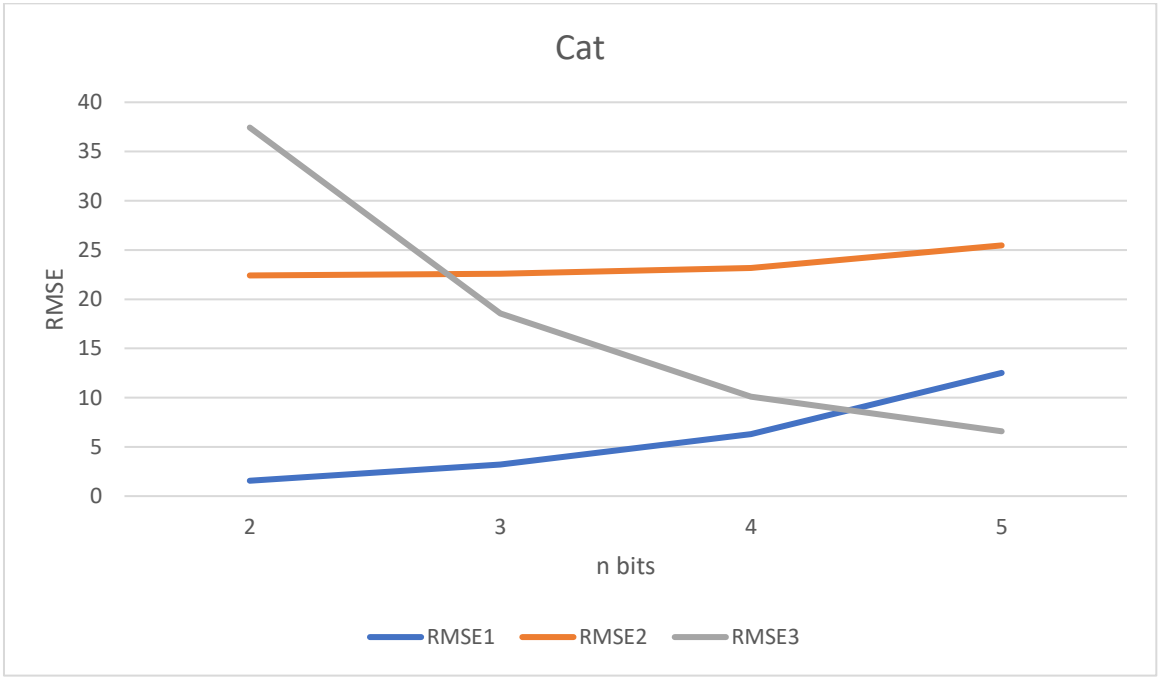
Additionally, final images with more n bits have closer values to white. For instance, if we keep 2 bits, we reach maximum 192, which is not close 255. For each bits the value we can have max:

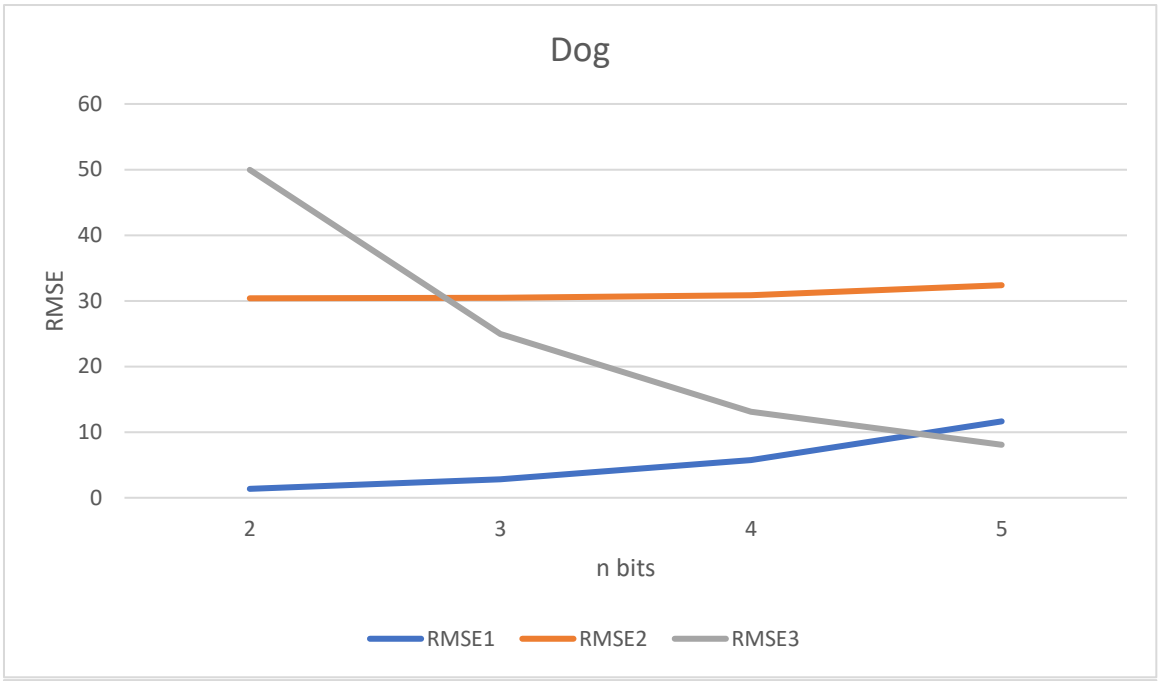
2: 192

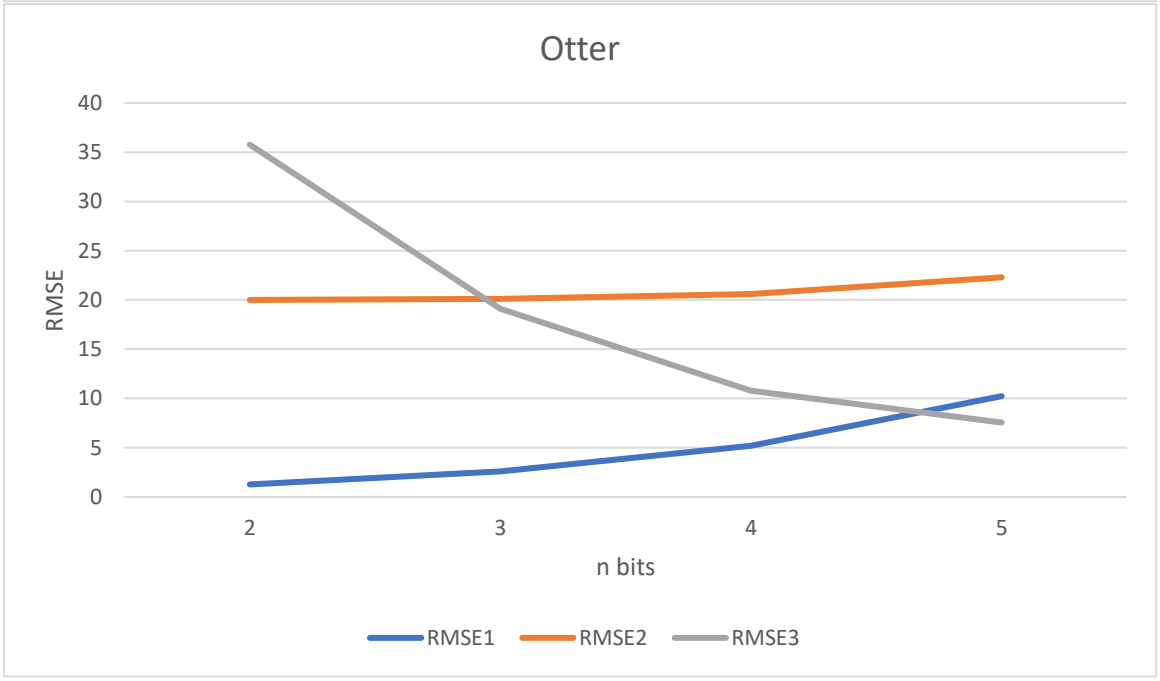
3: 224

4: 240

5: 248







I put the images from increasing n order for every three images below.