# algorithms implementation

## Popularity

1. Go through every pixels of the image to insert all colors to a map with RGB.
2. Find the 256 most popular colors in the map and build the color table
3. Find the proper color using the indexed color by (R-ri)\*(R-ri) + (G-gi)\*(G-gi) + (B-bi)\*(B-bi)

## Partition

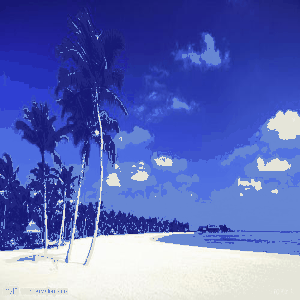
1. Go through every pixels of the image to find the min color range box.
2. Split the box to slices and build the color table
3. Find the color corresponding to the indexed color table

# Result

The original image



The image using Popularity



The image using Partition



The disadvantage of the popularity algorithm is that it completely throws out colors that appear infrequently. The most space of the image is sky which is blue. The color of the tree in the image using Popularity is losing the its color and using the color of the sky.

The disadvantage of uniform partitioning is that it does not account for the fact that the equal-sized partitions of the color space may not be equally populated. There may be many colors in one partition, and only a few in another. All the colors in a heavily populated populated will be converted to a single color, and smooth transitions of color will be lost in the image. We can see the color changed in the sky strangely.

The following is another image example, we can see the same disadvantage of the two algorithms.

The original image



The image using Popularity



The image using Partition

