

Informe de análisis y diseño

C++

juan Pablo Davila Bedoya

Departamento de Ingeniería Electrónica y
Telecomunicaciones
Universidad de Antioquia
Medellín
abril del 2021

1 Desarrollo

En principio el problema parece reducirse a encontrar alguna manera de acumular en 1 “pixel” de la imagen reescalada un color indicado, se supone que dicho color es resultado de una “ponderación de colores” extraído de un grupo de pixeles de la imagen original; es decir, vamos a suponer que recibimos una imagen (imgn1) con cierto numero de pixeles X , y necesitamos representar dicha imagen con Y pixeles (imgn2), siendo Y menor a X , tenemos 2 tareas entonces. La primera es calcular la cantidad de pixeles que contendrá un denominado “paquete”, con “paquete” nos referiremos a un grupo de pixeles de imgn1, por ejemplo un grupo de 4,6,8,...,n pixeles, imgn1 se seccionara entonces en n paquetes de pixeles, puede verse análogamente a ordenar un grupo de personas en un numero de sillas menor al numero de personas, para hacer esto, se tendrán que agrupar las personas en “paquetes de personas”, y asignar cada “paquete de personas” 1 sola silla. Cada 1 de los paquetes que tenemos ahora están conformados por una variedad de colores, sin embargo cada uno de estos paquetes representara 1 SOLO PIXEL de img2, por lo tanto cada paquete de colores tendrán que convertirse en un solo color, en teoría hacer esto es mucho mas sencillo que lo primero, se supone que esto se lograría midiendo los colores del paquete, y analizando cual es el color que mas se repite, ¡y listo!, ese color debería ser el color del nuevo pixel de imgn2 extraído de imgn1; solo queda repetir esto una cantidad de veces igual al numero de paquetes y wala!, abemus imagen reescalada.

Resumiendo los pasos serian los siguientes:

1. seccionar la imagen original en paquetes de n pixeles (cada paquete es 1 pixel de la imagen a formar).
2. A cada paquete asignarle un color como se explicó anteriormente.
3. repetir el paso 2 a todos los paquetes.

Hay un detalle con el paso 1, y es que no explico como pienso calcular la cantidad de pixeles que tendrá cada paquete, y la verdad es que no se muy bien como aún, pero estoy bastante seguro que se hará así:

El tamaño en x (ancho) de cada paquetes resultara de la división del tamaño en x de imgn1 entre imgn2. El tamaño en y (alto) de cada paquete resultara igualmente de la división en y de imgn1 entre imgn2.