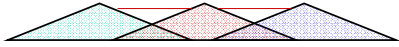
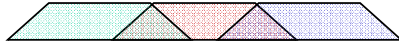


## Image feathering

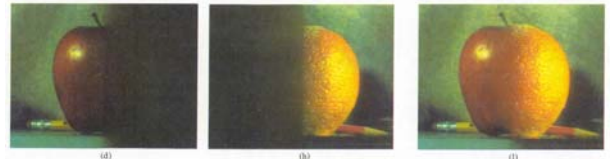
Weight each image proportional to its distance from the edge  
(distance map [Danielsson, CVGIP 1980])



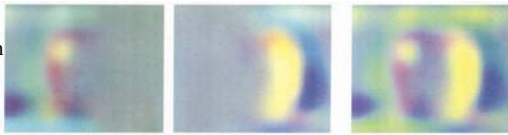
1. Generate *weight map* for each image
2. Sum up all of the weights and divide by sum:  
weights sum up to 1:  $w_i' = w_i / (\sum_i w_i)$



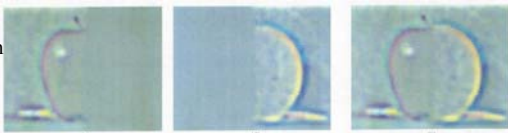
## Pyramid Blending



Laplacian  
level  
4



Laplacian  
level  
2



Laplacian  
level  
0



## Laplacian image blend

1. Compute Laplacian pyramid
2. Compute Gaussian pyramid on *weight* image (can put this in A channel)
3. Blend Laplacians using Gaussian blurred weights
4. Reconstruct the final image

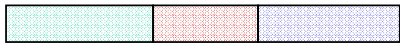
Q: How do we compute the original weights?

A: For horizontal panorama, use *mid-lines*

Q: How about for a general "3D" panorama?

## Weight selection (3D panorama)

Idea: use original feather weights to select *strongest* contributing image



Can be implemented using  $L^\infty$  norm: ( $p = 10$ )

$$w_i' = [w_i^p / (\sum_i w_i^p)]^{1/p}$$