Ashley Bourgoyne
CS 376

Problem Set 1

5.)

prague.jpg

a) original image

Ashley Bourgoyne
CS 376

Problem Set 1

b) seam carving resize

Ashley Bourgoyne
CS 376

Problem Set 1

c) matlab's resize

d) input: 480x640
output: 380x540
e) reduced width by 100, reduced height by 100
f) The boat stays well in tact because it is surrounded by rippling water, which creates many high energy points. The buildings get cut off and out of because they are surrounded by a very uniform sky. The leaves remain because they cause high energy areas as well.

Ashley Bourgoyne
CS 376

Problem Set 1

mall.jpg

a) original

Ashley Bourgoyne
CS 376

Problem Set 1

b) seam carving resize

Ashley Bourgoyne
CS 376

Problem Set 1

c) matlab's resize

d) input: 769x775
output: 569x675

e) reduced height by 200, reduced width by 100

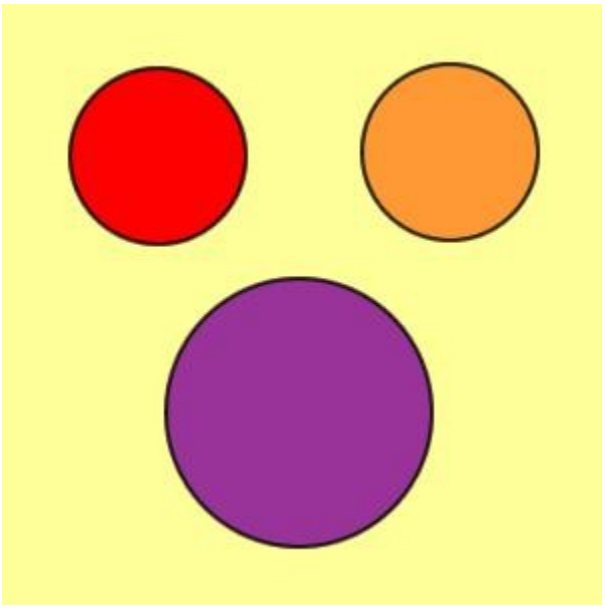
f) You can see the seams carved in the grass very easily, because, though the grass has low energy compared to the rest of the image, it is very textured. Also, the texture moves vertically, so since the seams where taken across this vertical movement it is very noticeable. Also, though the resize keeps the trees a the same proportion to the original image, it removes a large portion of the thickest trunk. This is because the tree continues from the top to almost the bottom of the image with very little change in energy vertically. A lot of the branches are thinned out because of this.

Ashley Bourgoyne
CS 376

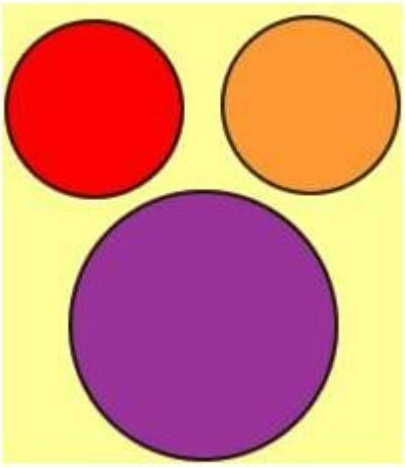
Problem Set 1

3circles.jpg (thesaurus.math.org)

a) original



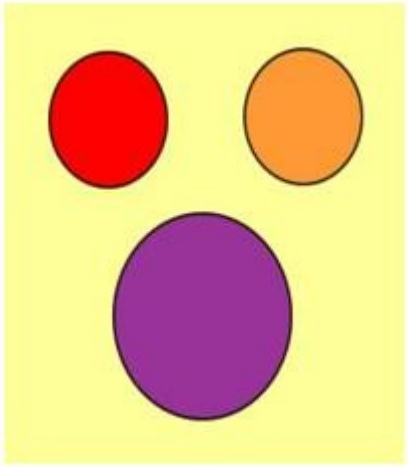
b) seam carving resize



Ashley Bourgoyne
CS 376

Problem Set 1

c) matlab's resize



d) input: 300x300
output: 230x200

e) reduced the height by 70, reduced the width by 100

f) The seam carving resize, unlike the matlab resize, preserves the shapes of the circles. This is because the area between them has no change in energy so the algorithm will automatically choose these seams first. This is an ideal example where seam carving can remove seams in between import areas of an image, while a simple sampling resize can not.

Ashley Bourgoyne
CS 376

Problem Set 1

mona-list.jpg (Leonardo da Vinci! Ninja turtle and Renaissance man)

a) original

Ashley Bourgoyne
CS 376

Problem Set 1

b) seam carving resize

c) matlab's resize

Ashley Bourgoyne
CS 376

Problem Set 1

d) input: 500x337
output: 300x337
e) reduced height by 200, reduced width by 100
f) Though this resize is not perfect, it's interesting to note how some of the features of this famous painting remained quite intact. Though Mona Lisa's hair was quite trimmed, and some of her fingers seems a little to slender, her face continues to appear human and similar to how Leonardo painted it, though some of the shading has disappeared. This is because the face is very detailed, causing it to have high energy. Even with it's defaults, it looks much better than the matlab resize. The landscape in the background also looks as if the oil paints remain untouched. This is due to the fact that Leonardo painted in a style called sfumato, where he made everything look gray and smoky. Therefore, the landscape blends well with each other even when seams are removed.

Ashley Bourgoyne
CS 376

Problem Set 1

winter.jpg (http://robinwoodchurch.wordpress.com/2010/08/16/surviving-the-deep-winter-of-the-church/)

a) original

Ashley Bourgoyne
CS 376

Problem Set 1

b) seam carving resize

c) matlab's resize

d) input: 375x500
output: 175x450

e) reduce width by 50 pixels, reduce height by 200

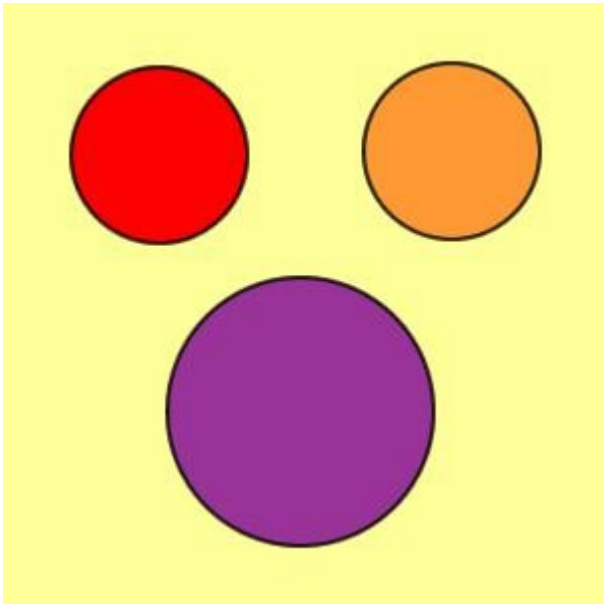
f) This example of seam carving worked quite well, keeping the major details in the picture. The mountains and trees are still to scale with each other, while the sky and snow bank have been diminished. This is because their energy moves in the vertical direction, creating high energy when trying to remove horizontal seams, or reduce the height. It looks much more realistic than the matlab resize which squashes the mountain and the trees, the two focuses of this picture.

Ashley Bourgoyne
CS 376

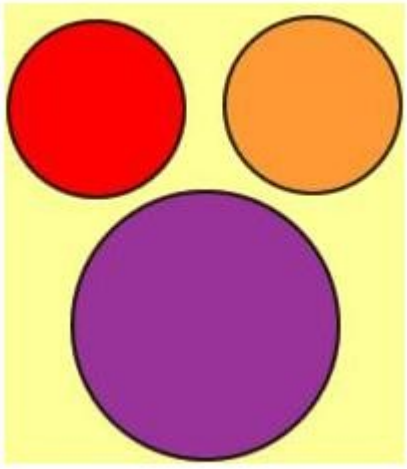
Problem Set 1

PART 111

5) original image



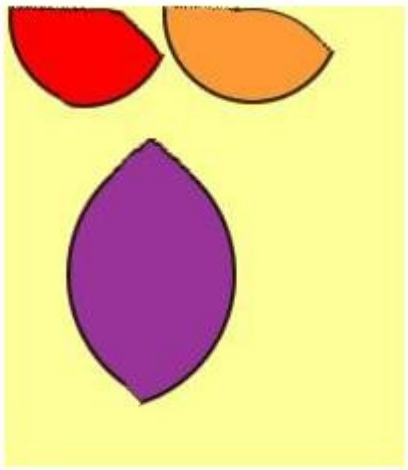
dynamic programming method (reduce height by 70, reduce width by 100)



Ashley Bourgoyne
CS 376

Problem Set 1

greedy method (reduce height by 70, reduce width by 100)



As you can see, the dynamic programming approach works much better than the greedy approach. This is because, using the greedy approach the method can not find the large gaps in between important features, like the dynamic one can. When picking vertical seems, once the initial yellow border is removed on the left, the next seem looks optimal, starting with a very low energy, but as it moves downwards it runs straight into a key feature of the image (a circle in this case). It would have been more wise to instead start in the middle and move around the circle, exactly as the dynamic method does.