Superpixel (SP): What & why?

'A **Superpixel** groups raw pixels into *perceptually meaningful atomic regions*'

- SP map is a more *natural* representation of a visual scene
  - Pixels are not natural entities
  - Pixel-grid is an inefficient representation
  - SPs provide richer feature descriptors (histograms, shape factors, spatial relationships etc.)
- Ideal SP map has some desirable properties [Ren, 2003]:
  1. Computationally efficient
  2. Representationally efficient
  3. Perceptually meaningful
  4. Near-complete

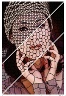


Figure: *Image segmented using SLIC into superpixels of size 64, 256, and 1024 pixels (approximately) [Achanta et al, 2012].*

1. i.e. a group of pixels which have similar characteristics. An atomic region is simply a single irreducible region of the larger image. SPs now an important preprocessing step in many vision systems.
2. Why use Superpixels instead of raw pixels?
3. Raw pixels are just an artefact of the digital image capture process.
4. Pixel-grid has a rigid structure. Pixel-grid is not a natural representation of visual scenes. Pixel-grid contains redundancy and is complex
5. An SP provides more information than a raw pixel
6. Definition of a good SP often depends on the application
7. **Computationally efficient**: SP reduce complexity of an image: $>$100k pixels to $<$1k superpixels. SP can be used as basic units (primitives) in the next image processing like segmentation, object detection.

Superpixel (SP): What & why?

'A **Superpixel** groups raw pixels into *perceptually meaningful atomic regions*'

- SP map is a more *natural* representation of a visual scene
  - Pixels are not natural entities
  - Pixel-grid is an inefficient representation
  - SPs provide richer feature descriptors (histograms, shape factors, spatial relationships etc.)
- Ideal SP map has some desirable properties [Ren, 2003]:
  1. Computationally efficient
  2. Representationally efficient
  3. Perceptually meaningful
  4. Near-complete

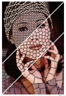


Figure: *Image segmented using SLIC into superpixels of size 64, 256, and 1024 pixels (approximately) [Achanta et al, 2012].*

1. **Representationally efficient**: Graphs over superpixels. Interactions between Superpixels occur over a much longer range that raw pixels. i.e. constructing graphs over the Superpixel representation of an image is much more efficient.
2. **Perceptually meaningful**: Individual raw pixels carry very little perceptual meaning. Pixels within Superpixel groupings share commonality, such as colour, texture etc.
3. **Near-complete**: Oversegmentation. Why oversegment? It's easier to merge SP, than split them. Most structures are conserved, important boundaries are found. There are few losses when mapping from pixels to SPs, since most SP algorithms oversegment.
4. The figure displays Superpixel maps (SLIC algorithm) for increasing Superpixel size overlaying the original image. The Superpixels are compact and their boundaries correspond to the natural boundaries of objects in the image. Insignificant boundaries are present, but SPs are easier to merge than split.

SP methods

- There are *many* methods for generating Superpixels
- Some are better suited to particular applications than others
- Desirable SP properties:
  1. Adhere well to image boundaries
  2. Generated as efficiently as possible
  3. Do not negatively impact performance of subsequent steps

Graph-based	**Gradient-ascent**
Pixels as node in a graph	Start from an initial clusters
Distance of pixel from centroid feature	Iteratively refine clusters
Edge weights represent affinities between pixel pairs	Cluster on pixel colour and/or Cartesian coordinates
Minimise a cost function over graph	SP formed when convergence criterion met

Table: *Two broad categories of SP methods*

1. There are 2 broad categories of SP methods.
2. Graph-based: Pixels as nodes in graph. Edge weights represent e.g. pairwise brightness, color and texture affinities between pixels.
3. Gradient-ascent: Iteratively refine from initial clusters of pixels until convergence criterion is met.

Agglomerative Clustering - [Felzenszwalb et al, 2004]

- Performs an *agglomerative clustering* of pixels as nodes on a graph
- Each pixel starts in its own segment
- Algorithm merges segments based on evidence for a boundary between two segments
- Adheres well to boundaries in practice
- Produces SP with irregular shapes and sizes
- No explicit control over SP quantity or their compactness

Figure: *Graph-based image segmentation.*

1. **Agglomerative clustering**: graph based approach. Bottom up approach where each pixel starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy.
2. Has the ability to preserve detail in low-variability image regions while ignoring detail in high-variability regions

Quick Shift - [Vedaldi et al, 2008]

- A fast *mode-seeking* segmentation scheme
- Associate each pixel to a mode of an underlying p.d.f.
- Identifies pixel clusters in joint spatial and colour dimensions
  1. Estimate a density for each pixel
  2. Follow the gradient of the density to assign each pixel to a mode
  3. Modes represent the final clusters

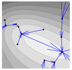


Figure: *Quick shift: Assignment of data points to mode using underlying density [Vedaldi et al, 2008].*

1. **Mode seeking algorithm** (like mean shift) which instead of iteratively shifting each point towards a local mean instead forms a tree of links to the nearest neighbor which increases the density.
2. **Medoids** are representative objects of a data set or a cluster with a data set whose average dissimilarity to all the objects in the cluster is minimal.

SLIC - [Achanta et al, 2012]

Figure: *Example images segmented using SLIC.*

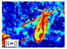
- It is a faster adaptation of k means clustering
- Initially k cluster centers are sampled on a regular grid
- Cluster centers are moved to avoid centering a SP on an edge
- Each pixel is associated with the nearest cluster center whose search region overlaps its location
- Cluster centers are adjusted to the mean vector of all the pixels belonging to the cluster
- Iterate the assignment and update steps until the error converges
- Disjoint pixels can be reassigned to nearby SP to enforce connectivity

1. k is the desired number of approximately equally sized SP. The cluster centers are the same amount of pixels apart.
2. Also to reduce the chance of seeding a SP with a noisy pixel. Moved to the lowest gradient position in a 3x3 neighbourhood.
3. D, a distance measure, determines the nearest cluster center for each pixel. This is the key to speeding up the algorithm because limiting the size of the search region significantly reduces the number of distance calculations, and results in a significant speed advantage over conventional k-means clustering where each pixel must be compared with all cluster centers.
4. However 10 iterations suffices for most images

(a) Depth estimation - [Zitnick & Kang, 2007]

(b) Segmentation - [Li et al, 2004]

(c) Body model estimation - [Mori, 2005]



(d) Object localisation - [Fulkerson et al, 2009]

1. Using SP rather than pixels provides robustness to noise and intensity bias when computing depths of a segment. Uses a simple K-means technique for clustering neighbour colours
2. A graph based algorithm is used to make SP so that the foreground of an image can easily be cut from the background
3. Using SP as a preprocessing paradigm improves the efficiency and accuracy of model search in an image
4. Identify and localise object classes in images. The quick shift algorithm is used to produced SP. A classifier is constructed on the histogram of local features found in each SP. Neighbourhoods of SP are aggregated to create a robust region classifier. Dark red means the image is foreground, while dark blue means its background. N is the number of neighbours considered.