Segmentation

STUDENTS-HUB.com
Uploaded By: anonymous

Outline

- Element of Image Analysis
- Applications
- Classical Segmentation Techniques
  - Edge-based
  - Thresholding-based
  - Region Based
  - Clustering Based
  - Graph Based
  - Others
- Deep Learning Techniques
  - Semantic Segmentation
  - Instance Segmentation

STUDENTS-HUB.com

Uploaded By: anonymous

Element of Image Analysis

Preprocess

Image acquisition, enhancement and restoration

Intermediate process

Image segmentation & Feature extraction

High level process

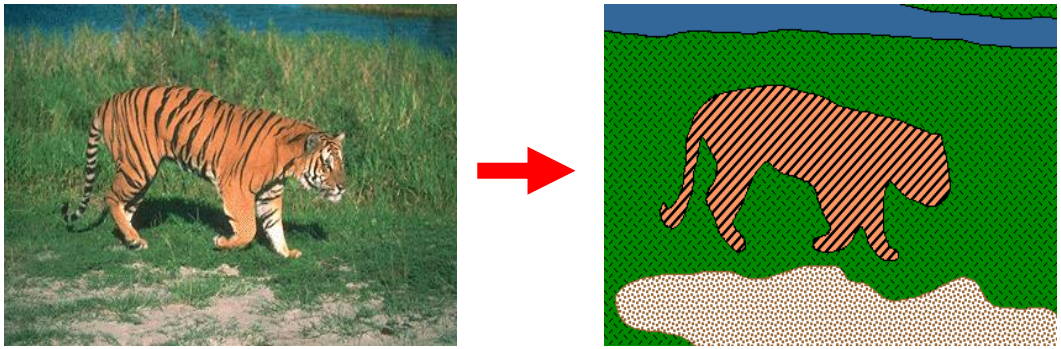
Image interpretation and recognition

STUDENTS-HUB.com

Uploaded By: anonymous

Image Segmentation

- Image segmentation is the operation of partitioning an image into a collection of connected sets of pixels



Segment image into:

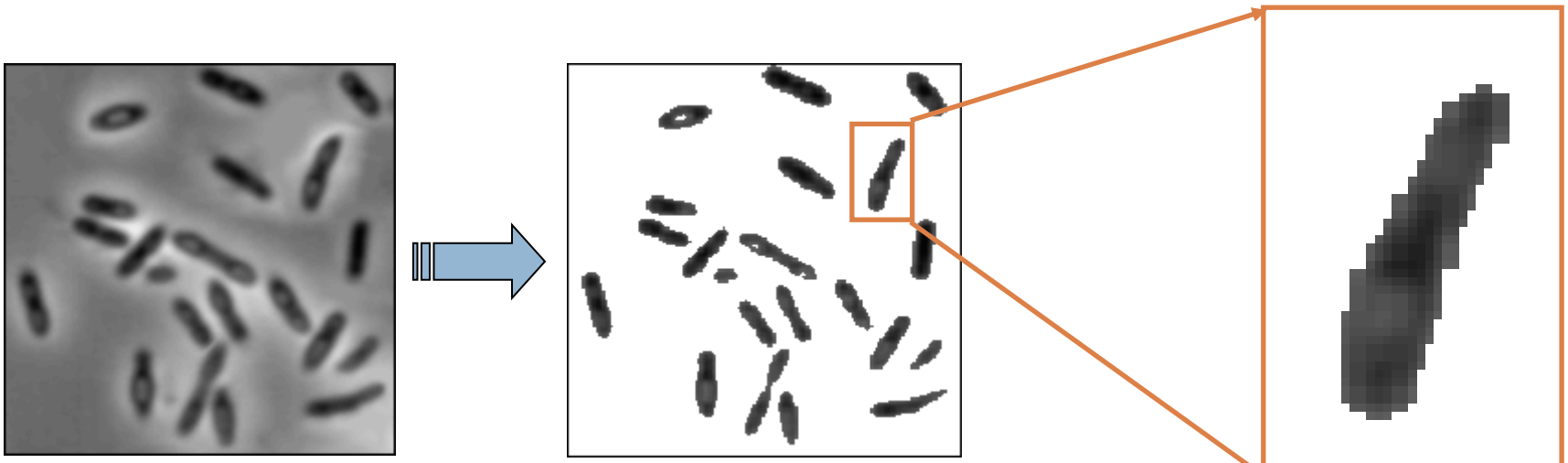
1. Meaningful regions (coherent objects)
2. Linear structures (line, curve,…)
3. Shapes (circles, ellipses, …)

STUDENTS-HUB.com

Uploaded By: anonymous

Why Image Segmentation

- Importance of Image Segmentation
  - Image segmentation is used to separate an image into constituent parts based on some image attributes.
  - Image segmentation is an important step in image analysis:
    1. Image segmentation reduces huge amount of unnecessary data while retaining only importance data for image analysis
    2. Image segmentation converts bitmap data into better structured data which is easier to be interpreted

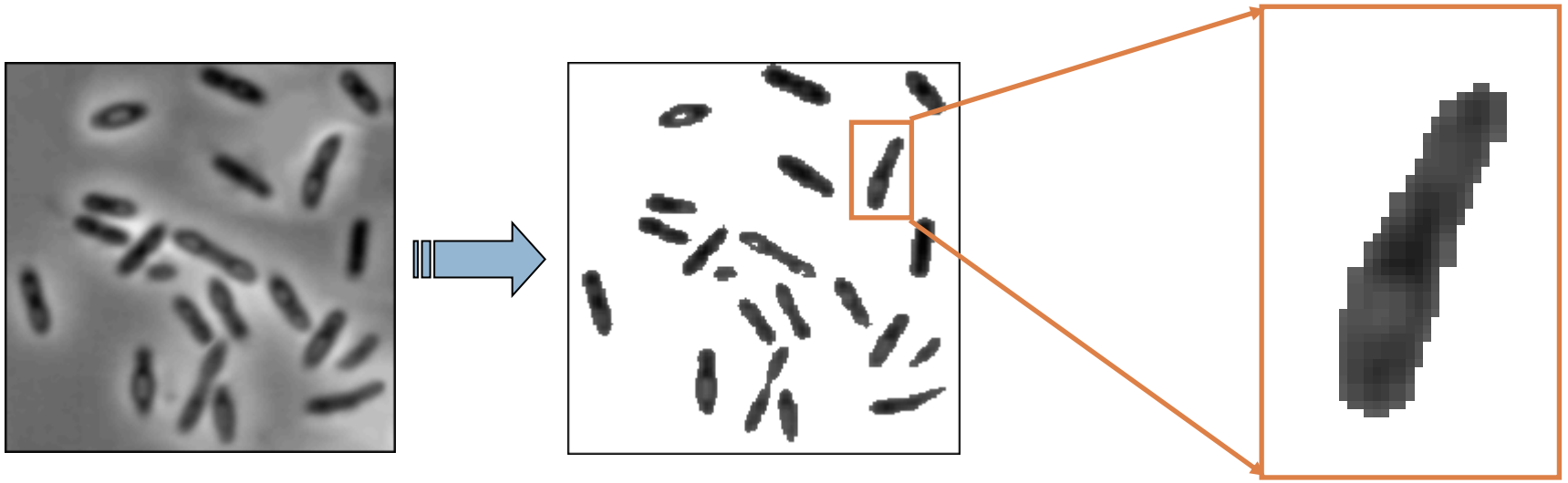


STUDENTS-HUB.com

Uploaded By: anonymous

From images to objects

- Goal of segmentation
  - Separate image into coherent “objects”
- Image segmentation is an important step in image analysis:
  1. Image segmentation reduces huge amount of unnecessary data while retaining only importance data for image analysis
  2. Image segmentation converts bitmap data into better structured data which is easier to be interpreted

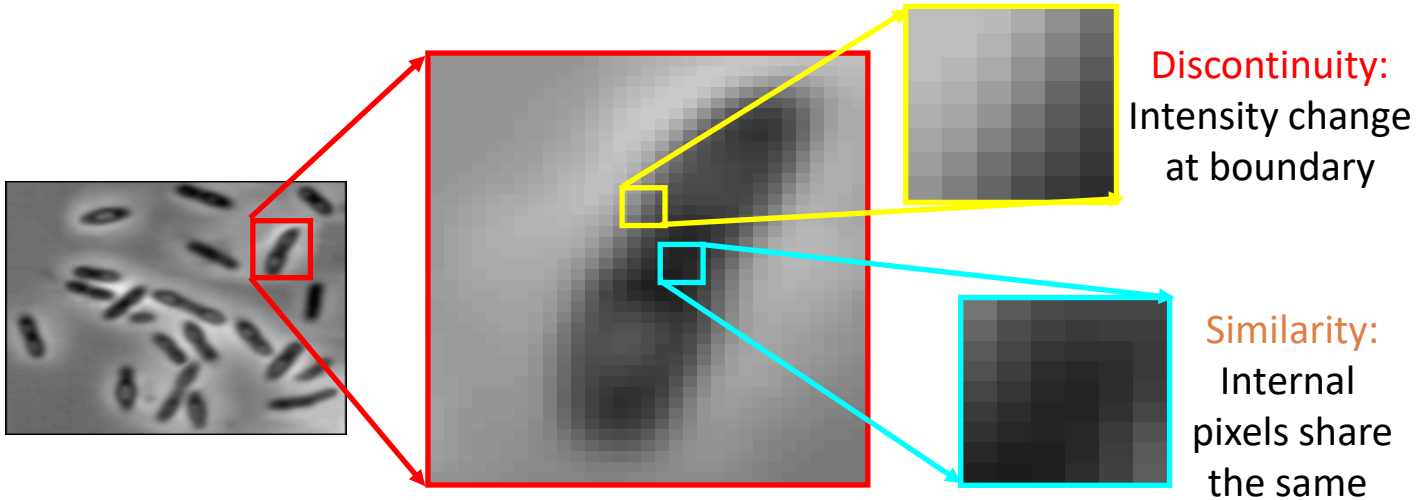


STUDENTS-HUB.com

Uploaded By: anonymous

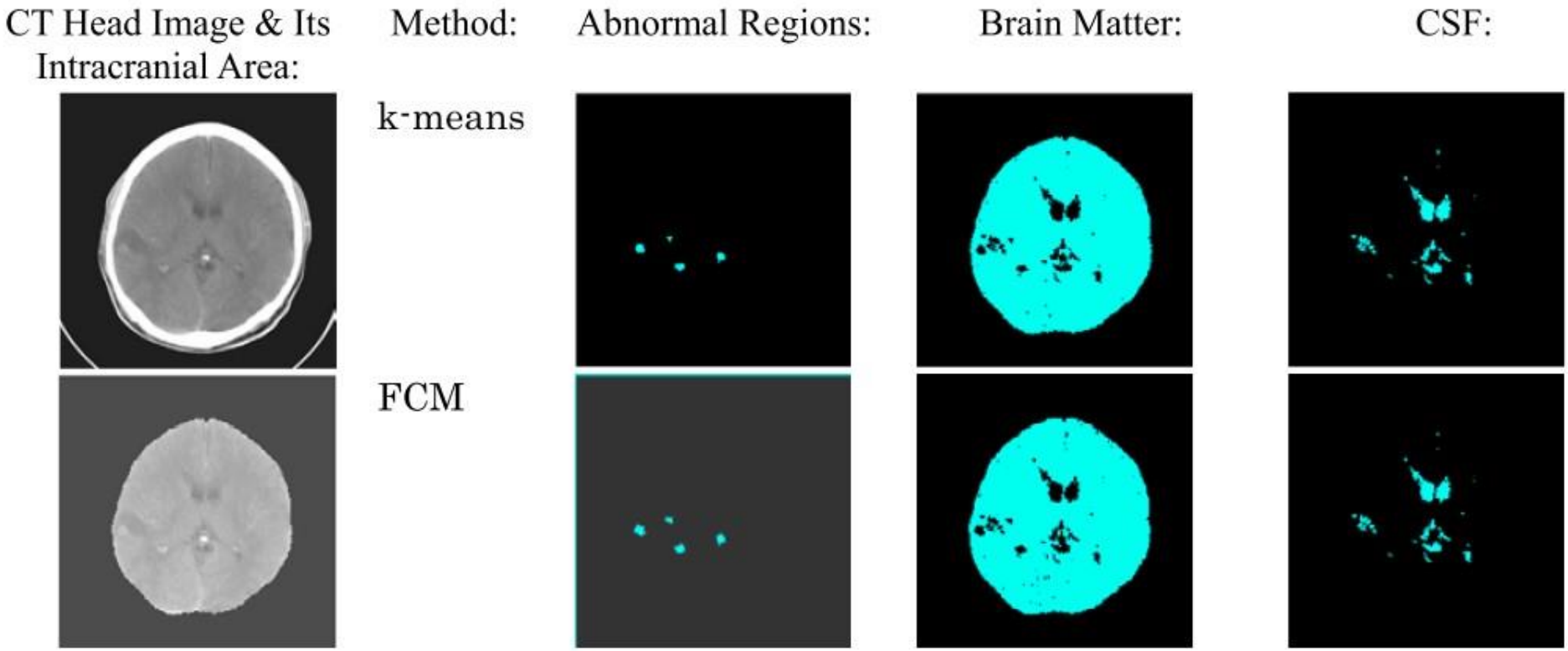
Fundamentals

- What defines an object?
  - Subjective problem, but has been well-studied.
  - Gestalt laws seek to formalize this.
  - "What is interesting and what is not" depends on the application.
  - Broad theory is absent at present.
- Image Attributes for Image Segmentation
  1. Similarity properties of pixels inside the object are used to group pixels into the same set.
  2. Discontinuity of pixel properties at the boundary between object and background is used to distinguish between pixels belonging to the object and those of background.



Applications of Image Segmentation

- Medical Image Analysis



Applications of Image Segmentation

- Remote Sensing Image Analysis

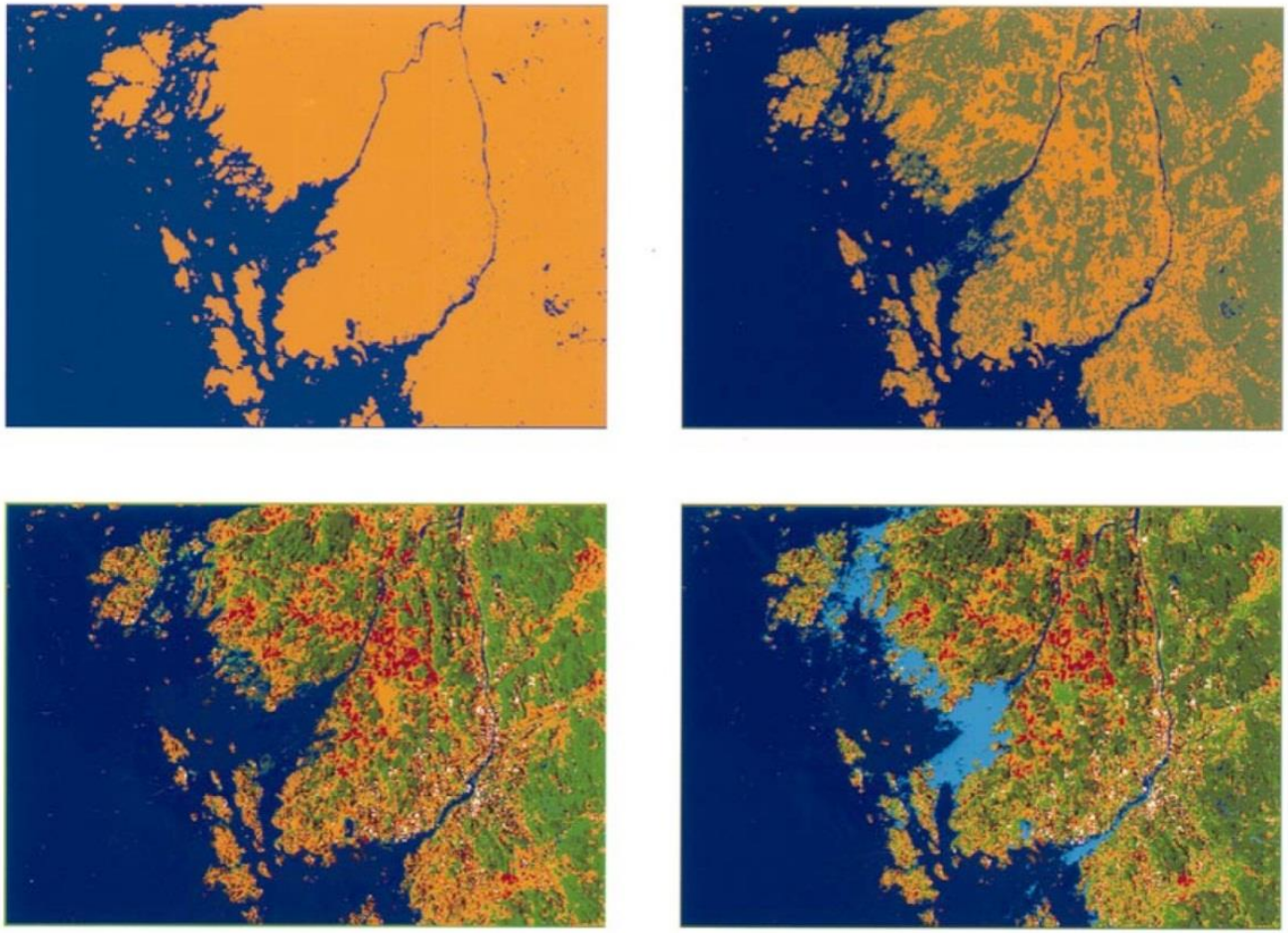


Fig. 7. Four clustering results for Gothenburg, number of clusters are two (top left), three (top right), nine (bottom left), and ten (bottom right). Shades of blue: water; shades of green: forests; white: man-made structures (urban); orange: open areas (other); red: farmlands; and black: system effects (foreshortening and layover). The light blue is the sea ice from the winter images.

STUDENTS-HUB.com

Uploaded By: anonymous

Applications of Image Segmentation

- Document Analysis – Background removal

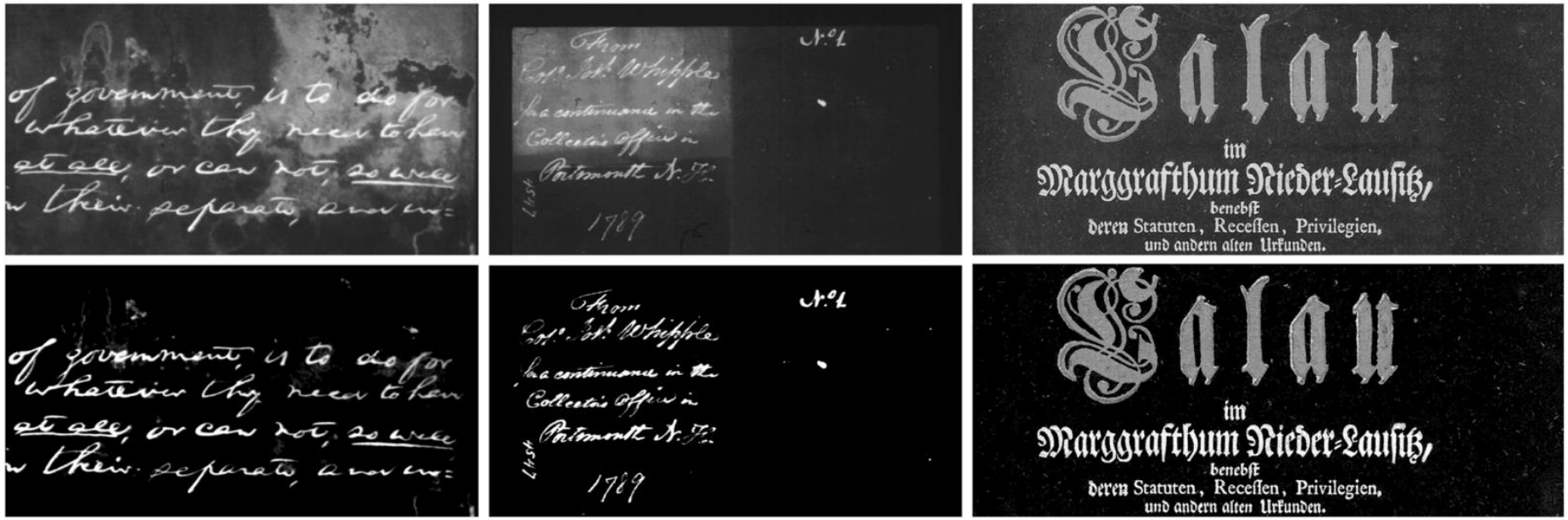


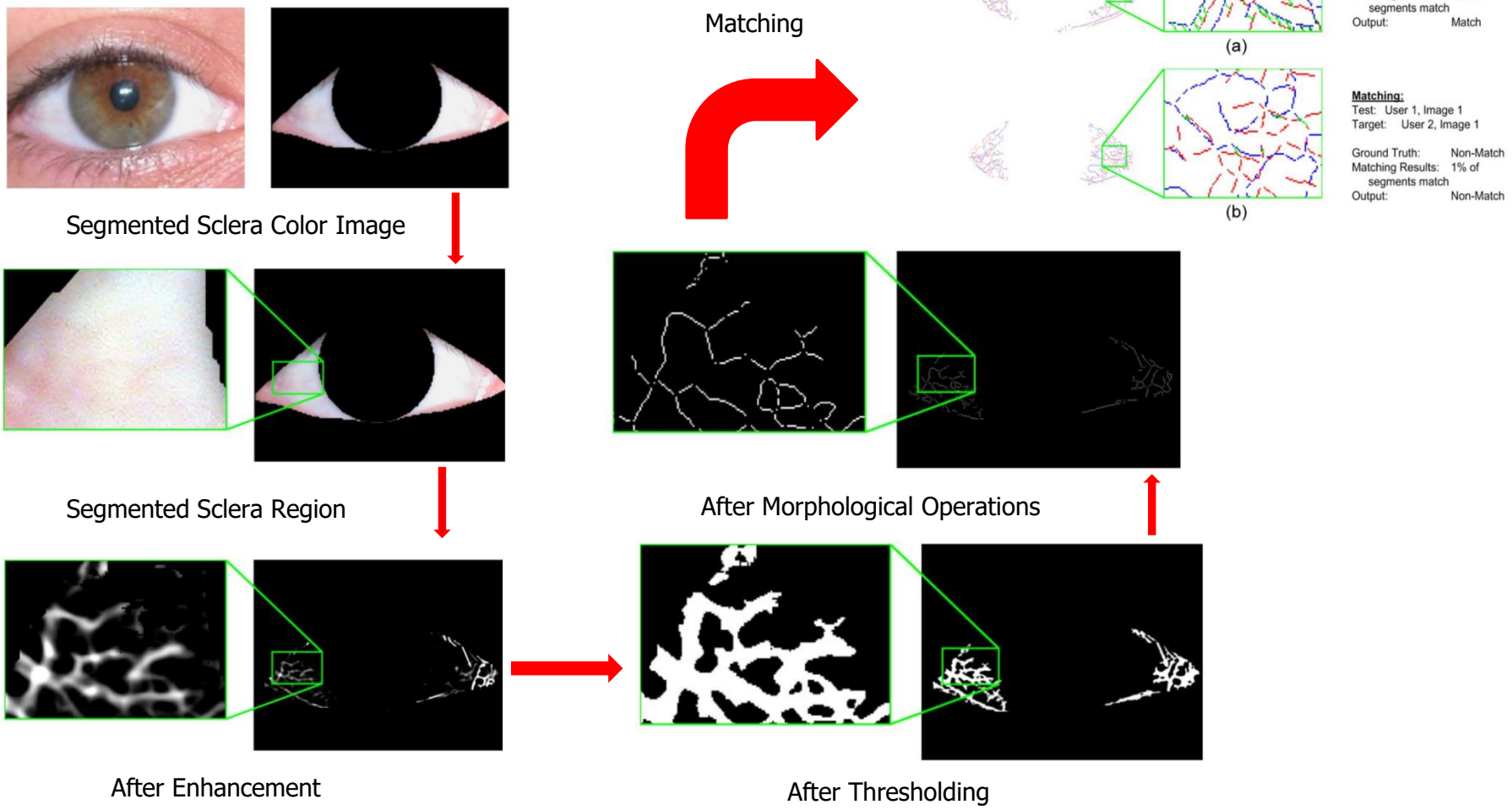
Fig. 11. Background removal based on h-component tree processing. (Top) Original document images and (bottom) corresponding results of the proposed background removal method.

STUDENTS-HUB.com

Uploaded By: anonymous

Applications of Image Segmentation

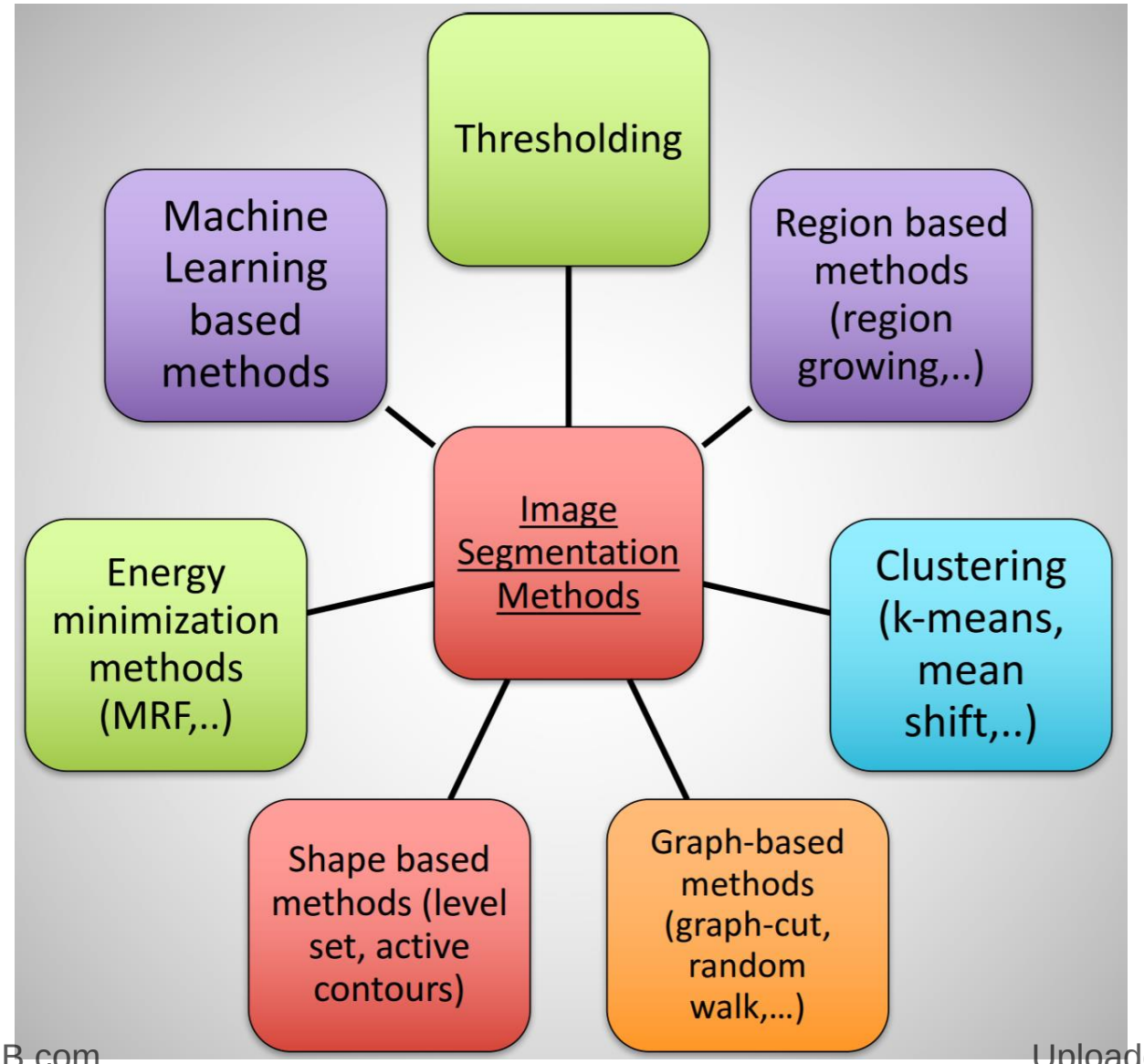
- Sclera Recognition as human identification



STUDENTS-HUB.com

Uploaded By: anonymous

Image Segmentation Techniques



STUDENTS-HUB.com
Uploaded By: anonymous

Image Segmentation Techniques

- **Edge-based methods**
  - In this approach, the object's boundaries/edges are used to detect the object in an image.
- **Thresholding based**
  - Thresholding is one of the simplest and earliest image segmentation techniques. It is used to binarize an image into a white-black image, based on the assumption that the object's entity can be detected using the intensity values of its pixels.
- **Region based**
  - Region-based methods are based on grouping pixels together according to a predefined criterion. The criterion can be based on pixel intensity, grayscale, or even color.
- **Clustering-Based:**
  - Group pixel with similar features (e.g. color, intensity, location) into one cluster without considering spatial relationship. The simplest case of clustering is binarizing a given image, which basically segmentation it into two classes, using a single threshold value.

STUDENTS-HUB.com Uploaded By: anonymous

Image Segmentation Techniques

- **Graph based**
  - Consider image as a weighted graph where pixels are represented by nodes and each edge has the corresponding nonnegative weight *which represents* similarity between the neighboring pixels. Partitioning *nodes* into two disjoint sets A and B where $A \cup B = V$ and $A \cap B = \emptyset$.
- **Shape based**
  - Adjust the segmentation process using a prior object shape model
- **Learning based**
  - Supervised based (Classification): the segmentation task can be viewed as a classification problem by classifying each pixel into different predefined categories.

STUDENTS-HUB.com Uploaded By: anonymous

Image Segmentation Techniques

- **Learning based**
  - Supervised based (Classification): the segmentation task can be viewed as a classification problem by classifying each pixel into different predefined categories.
  - Unsupervised (Clustering): try to group pixel with similar features (e.g. color, intensity, location) into one cluster without considering spatial relationship. The simplest case of clustering is binarizing a given image, which basically segmentation it into two classes, using a single threshold value.
    - K-Mean, Hierarchal clustering….
    - Mean shift

STUDENTS-HUB.com Uploaded By: anonymous

Outline

- Element of Image Analysis
- Applications
- Classical Segmentation Techniques
  - Edge-based
  - Thresholding-based
  - Region-based
  - Clustering Based
  - Graph Based
  - Others
- Deep Learning Techniques
  - Semantic Segmentation
  - Instance Segmentation

STUDENTS-HUB.com

Uploaded By: anonymous

Segmentation by Region Growing

- Region growing is a procedure that groups pixels or sub regions into larger regions.
- A simple approach to image segmentation is to start from some pixels (seeds) representing distinct image regions and to grow them, until they cover the entire image
- For region growing we need a rule describing a growth mechanism and a rule checking the homogeneity of the regions after each growth step
- The growth mechanism – at each stage k and for each region Ri(k), i = 1,…,N, we check if there are unclassified pixels in the 8-neighbourhood of each pixel of the region border
- Before assigning such a pixel x to a region Ri(k),we check if the region homogeneity:

  $P(Ri(k) \cup \{x\}) = TRUE$, is valid

STUDENTS-HUB.com Uploaded By: anonymous

Segmentation by Region Growing

- Homogeneity test: if the pixel intensity is close to the region mean value

  |I(r,c) – M(i)| <= T(i)

- Threshold Ti varies depending on the region Rn and the intensity of the pixel I(r,c).It can be chosen this way:

  T(i) = { 1 – [s.d(i)/M(i)] } T

- The arithmetic mean m and standard deviation *sd* of a class Ri having n pixels can be used to decide if the merging of the two regions R1,R2 is allowed, if

  |M1 – M2| < (k)s.d(i) , i = 1, 2 , two regions are merged

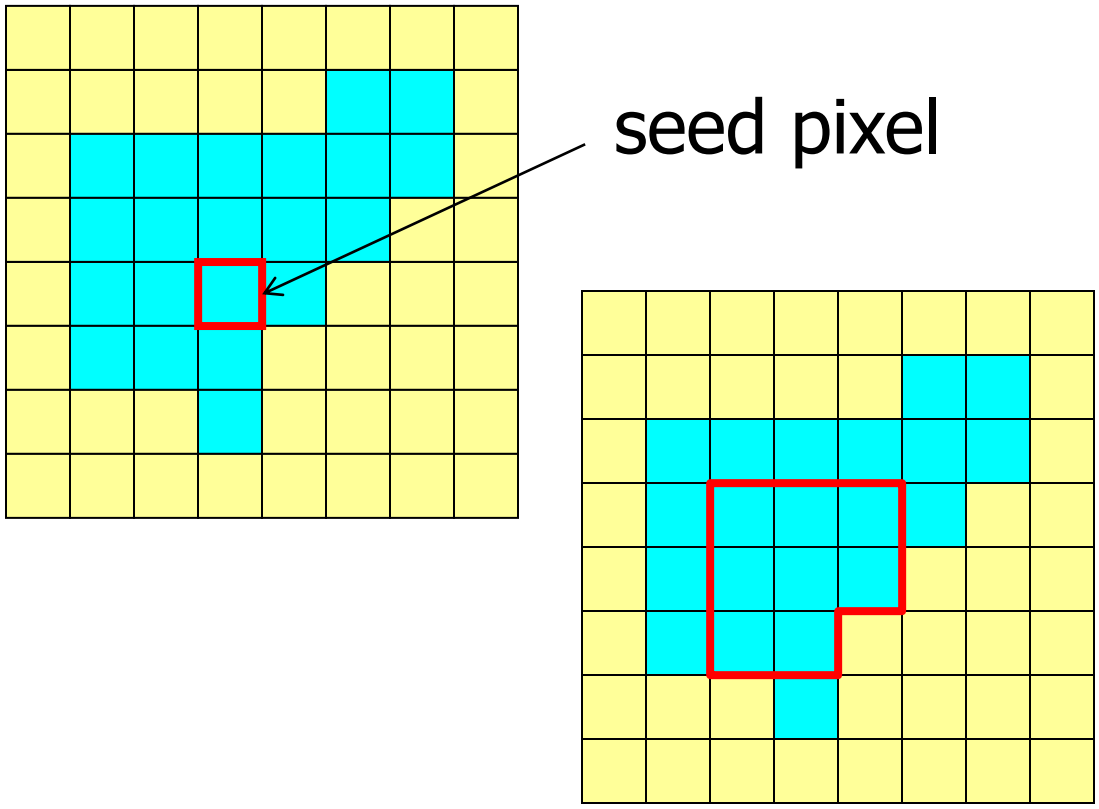
STUDENTS-HUB.com Uploaded By: anonymous

Segmentation by Region Growing

- **A General Region Growing Algorithm**
  1. Define the similarity criteria (difference intensity, variance, …)
  2. Define the stopping criteria (properties of the region, size, shape, …)
  3. Select a set of seed pixels; S
  4. Define an empty array O(x,y) and initialize its elements to 0's except at seed locations. Seed locations are assigned different intensity values
  5. For each seed in S, check its neighbors (4-,8-, or d-neighbors) against the similarity criteria
  6. If any of the neighboring pixels satisfy the criteria, then set the corresponding location in O to the same intensity level as its seed
  7. Check the if the stopping criteria is not met. If not, repeat steps 4 through 7 to the newly added pixels
  8. Repeat steps 4 through 7 until all seed pixels are processed

STUDENTS-HUB.com Uploaded By: anonymous

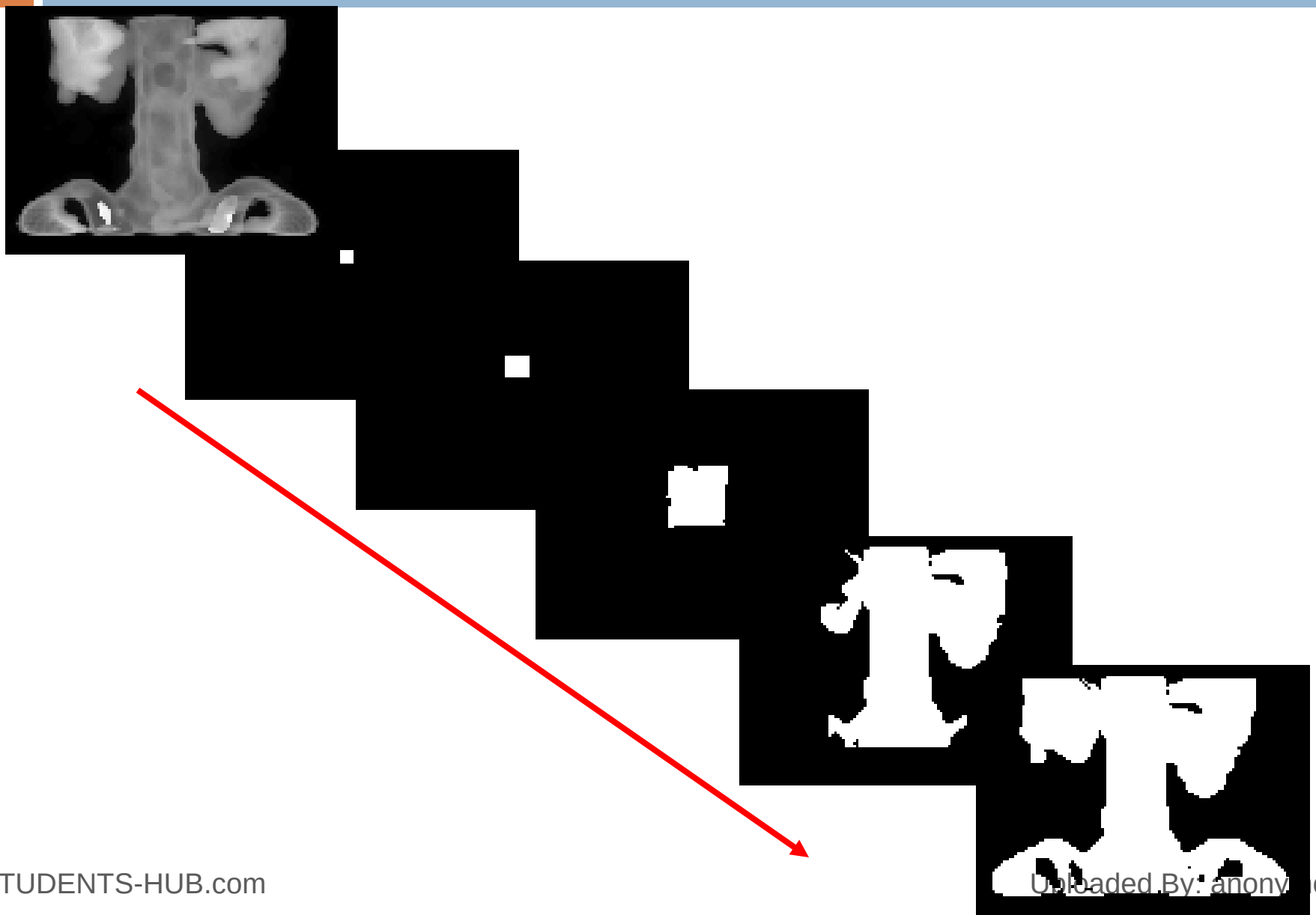
Segmentation by Region Growing



STUDENTS-HUB.com

Uploaded By: anonymous

Segmentation by Region Growing



STUDENTS-HUB.com

Uploaded By: anonymous

Segmentation by Region Growing

- How to select the seeds ?
  - Nature of problem
  - Random
  - Interactively

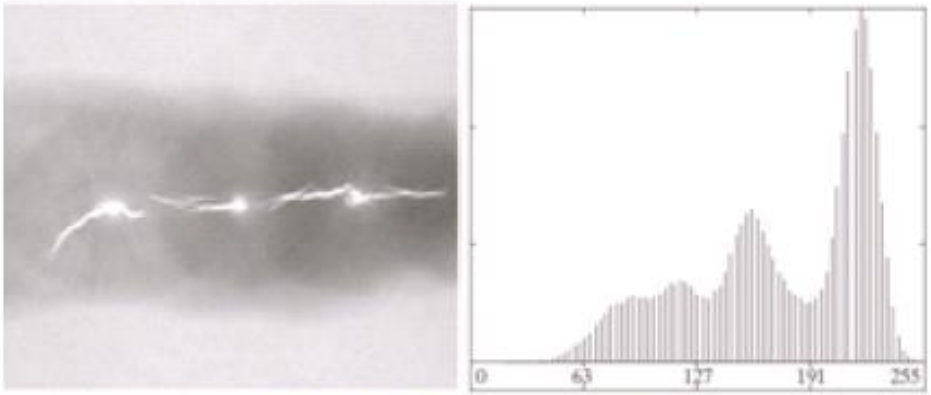
- How to select the similarity properties?
  - Nature of problem and image type (color ,monochrome ..)
  - Use statistical measures of local neighborhoods
  - May incorporate pixel location

STUDENTS-HUB.com

Uploaded By: anonymous

Segmentation by Region Growing

- **Example: Segment the cracks in the weld**



Original Image and its histogram

Thresholding result to identify regions of high intensity

Seeds specified by random selection from those in the thresholded image

Result of region growing

$$g(x,y) = \begin{cases} 1, & |f(x,y)-N(x,y)| > 126 \\ 0, & otherwise \end{cases}$$

STUDENTS-HUB.com Uploaded By: anonymous

Region growing segmentation: example

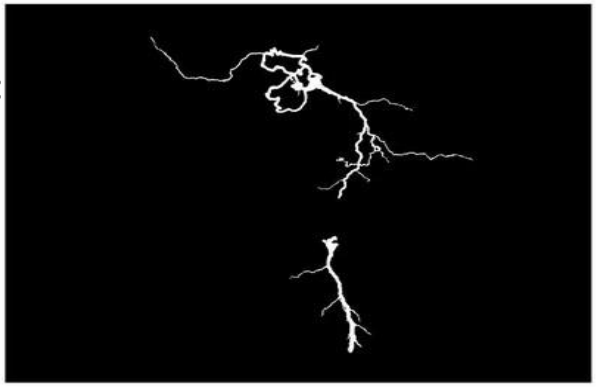
Problem: To isolate the strongest lightning region of the image on the right hand side without splitting it apart.

Solution: To choose the points having the highest gray-scale value which is 255 as the seed points shown in the image immediately below.

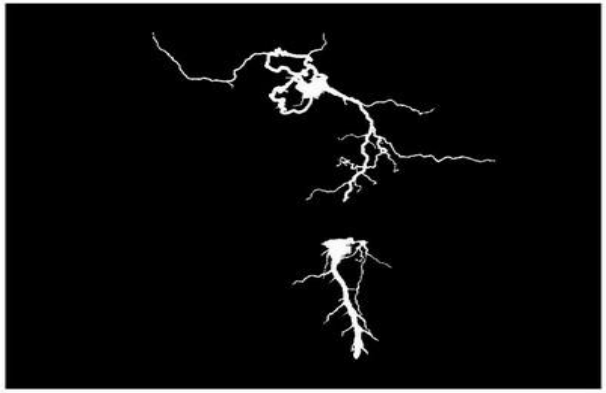
original image

threshold = 255 returns multiple seeds

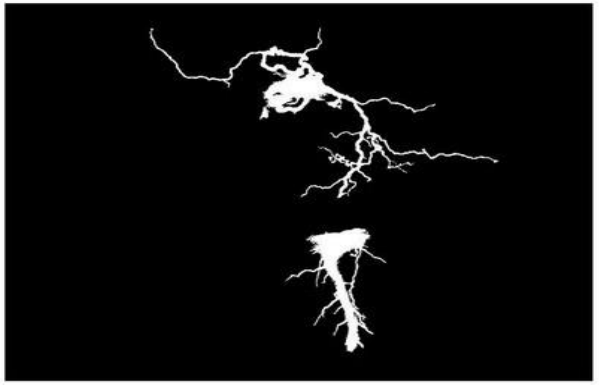
threshold: 225~255



threshold: 190~225



threshold: 155~255



STUDENTS-HUB.com

Uploaded By: anonymous

Region growing segmentation

- **Advantages:**
  - It is a fast method.
  - It is conceptually simple.

- **Disadvantages:**
  - Local method: no global view of the problem.
  - Gradient problem: in practice, there is almost always a continuous path of points related to color close that connects two points of an image. Thus, unless we use a pre-defined variance (threshold), this will lead to the gradient problem.
  - Algorithm very sensitive to noise.

STUDENTS-HUB.com Uploaded By: anonymous

Region splitting and merging segmentation

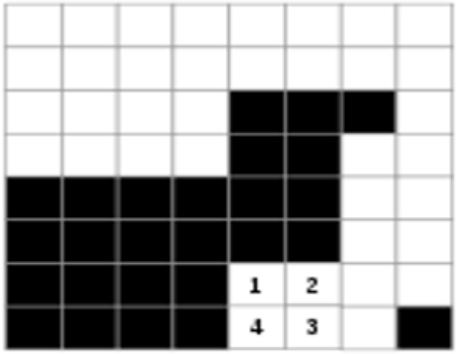
- **Region splitting:**
  - Unlike region growing which starts from a set of seed points, region splitting starts with the whole image as a single region and subdivides it into subsidiary regions recursively while a condition of homogeneity is not satisfied.

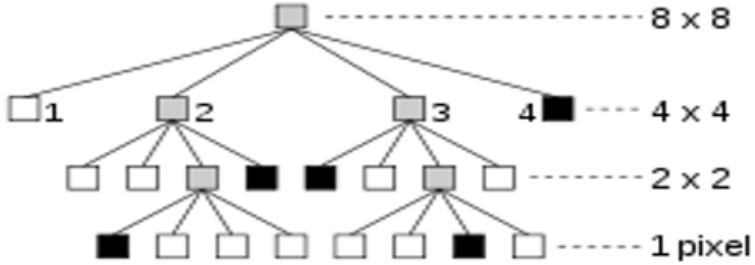
- **Region merging:**
  - Region merging is the opposite of region splitting, and works as a way of avoiding over-segmentation.
  - Start with small regions (e.g., 2x2 or 4x4 regions) and merge the regions that have similar characteristics (such as gray level, variance).

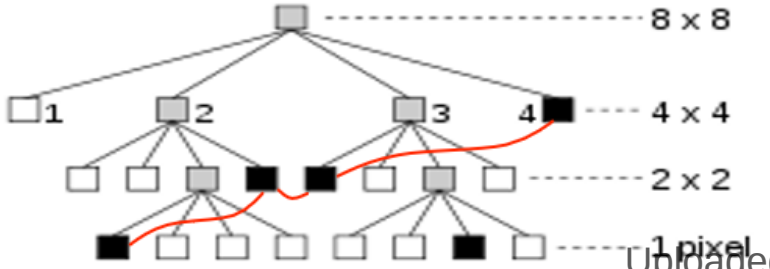
STUDENTS-HUB.com Uploaded By: anonymous

Splitting & merging segmentation algorithm

- **Two data structures:**
  - Quadtree for splitting:
    - Splitting is a top-down procedure that creates regions that may be adjacent and homogeneous, but not merged.
  - RAG (region adjacency graph) for splitting and merging:
    - Splitting and merging work together iteratively, i.e., at each iteration of quadtree partitioning.
    - RAG has an embedded quadtree for splitting that represents 4 containment relations.
    - RAG also represents 4 adjacency relations (one per square side).



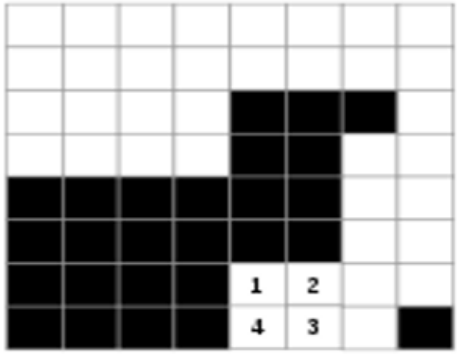


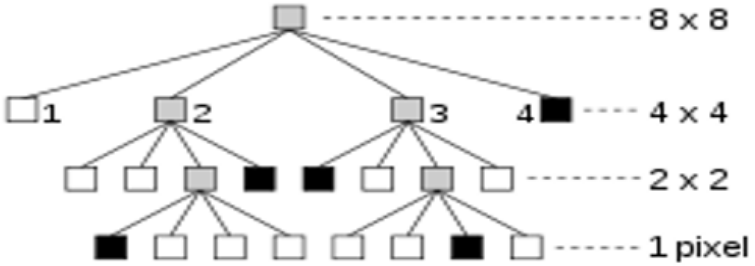


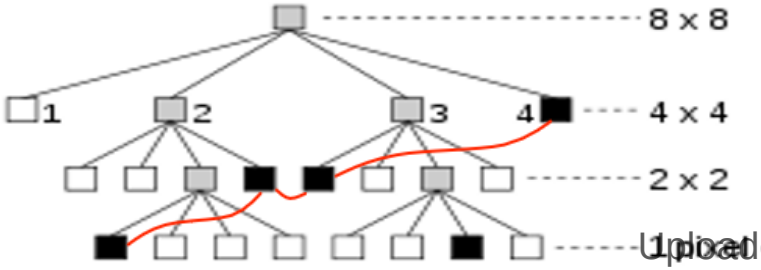
STUDENTS-HUB.com
Uploaded By: anonymous

Splitting & merging segmentation algorithm

- If a region R is inhomogeneous (P(R)=FALSE), then R is split into four sub-regions.
- If two adjacent regions R_1,R_2 are homogeneous ($P(R_1 \cup R_2)$=TRUE), they are then merged.
- The algorithm stops when no further splitting or merging is possible.
- If you have another image you'd like me to transcribe, feel free to share it!

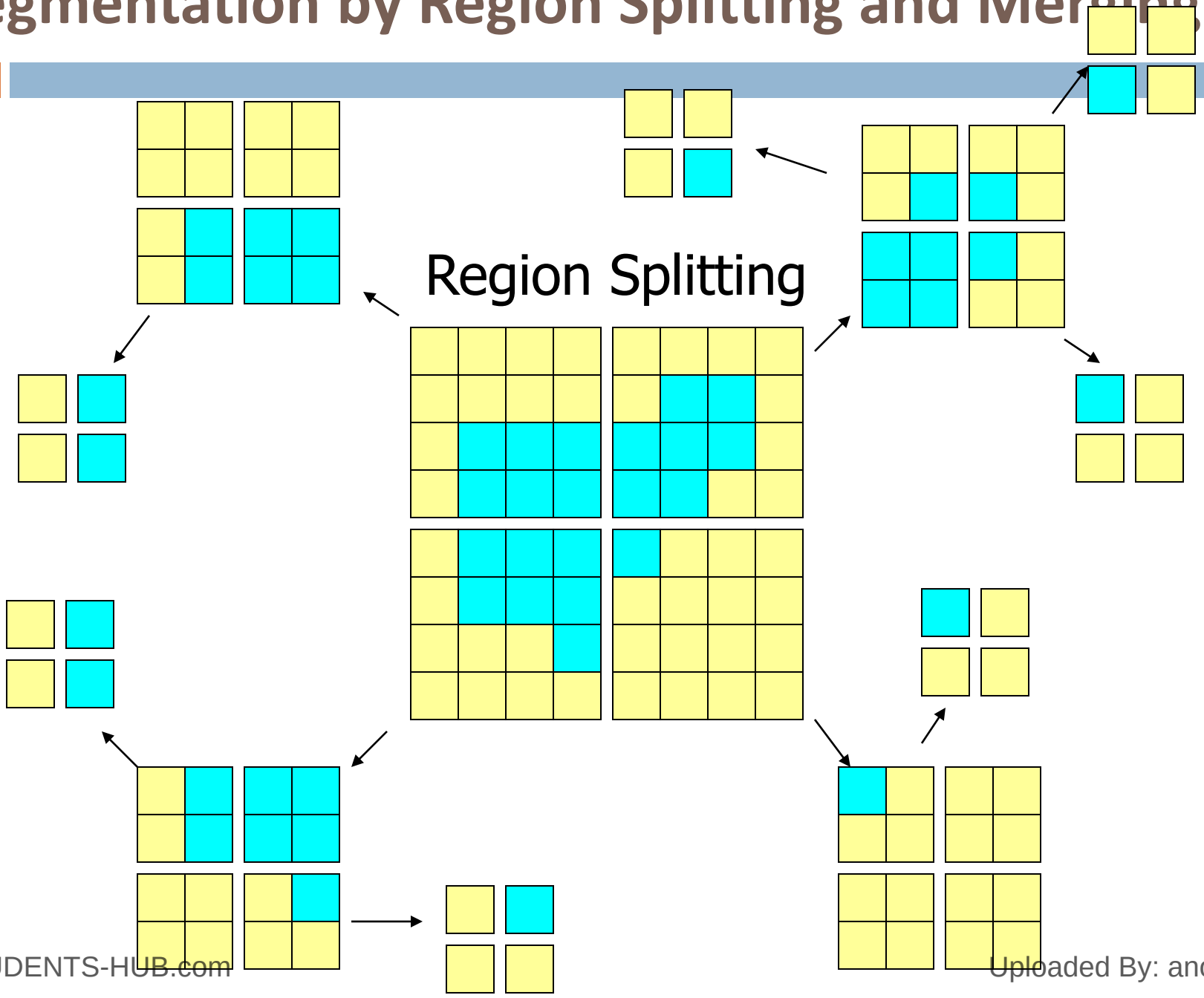






STUDENTS-HUB.com
Uploaded By: anonymous

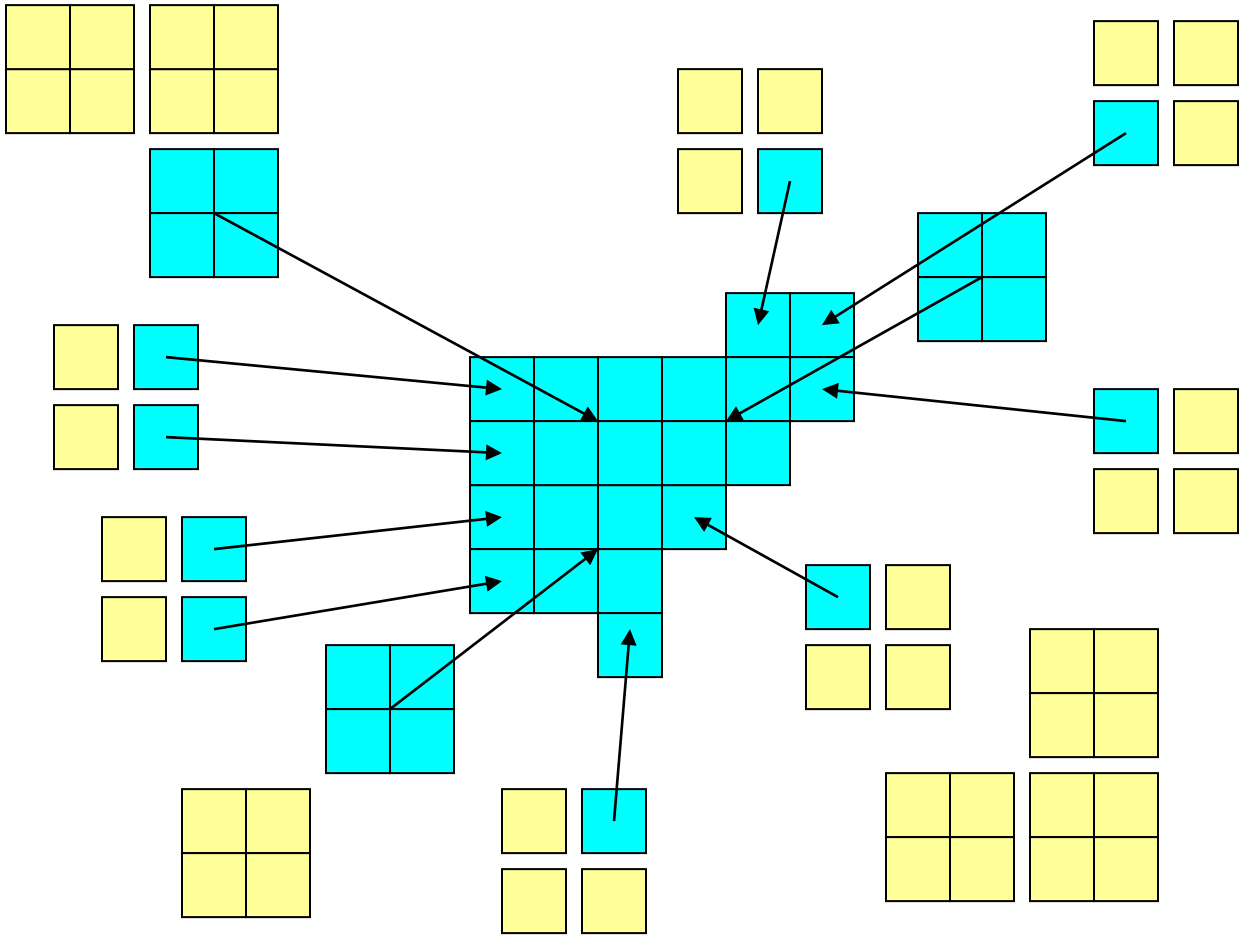
Segmentation by Region Splitting and Merging



STUDENTS-HUB.com
Uploaded By: anonymous

Segmentation by Region Splitting and Merging

Merging



STUDENTS-HUB.com

Uploaded By: anonymous

Region splitting: *example*

In this example, the **criterion of homogeneity** is the variance of 1.

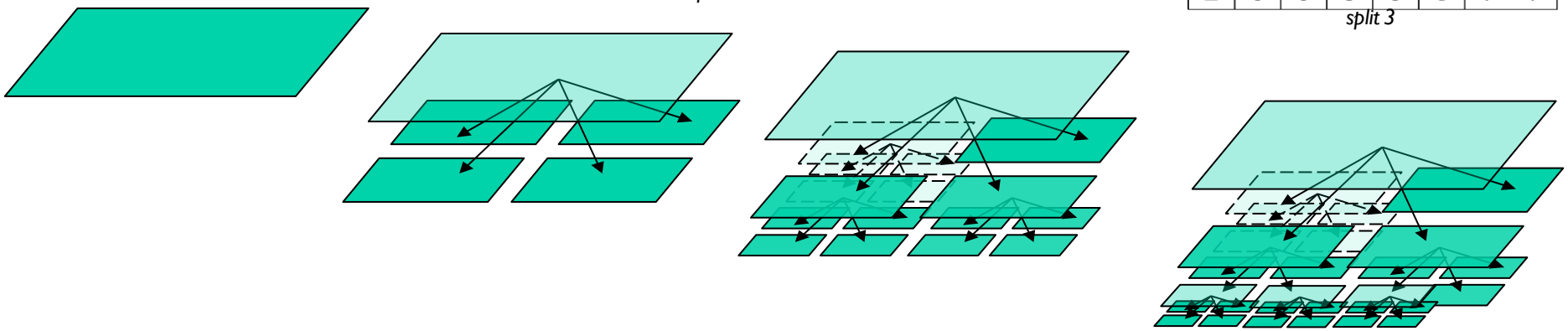
0	1	0	0	7	7	7	7
1	0	2	2	7	7	7	7
0	2	2	2	7	7	7	7
4	4	2	2	7	7	7	7
0	0	1	1	3	3	7	7
1	1	2	2	3	7	7	7
2	4	3	0	5	7	7	7
2	3	3	5	5	0	7	7

original image

split 1

split 2

split 3

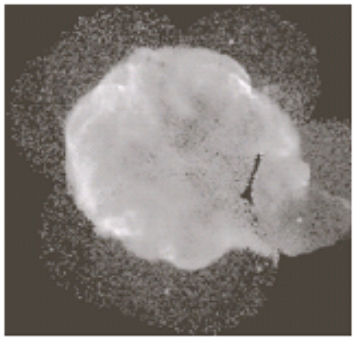


Uploaded By: anonymous

Segmentation by Region Splitting and Merging

- **Example**
  - Segment the less dense matter which is of noisy nature (high standard deviation when compared to background and the dense region) and moderate intensity
  - Use the predicate function

$$g(x,y) = \begin{cases} 1\ , & \sigma > 10 \text{ and } \ 0<m<126 \\ 0\ , & otherwise \end{cases}$$



Original

Minimum quadrant size 32x32

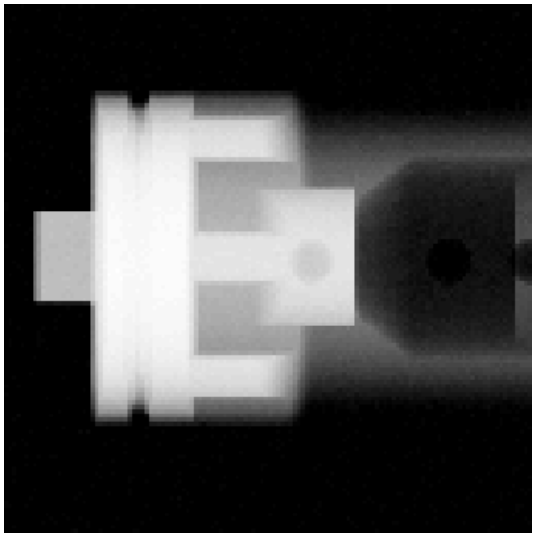
Minimum quadrant size 16x16

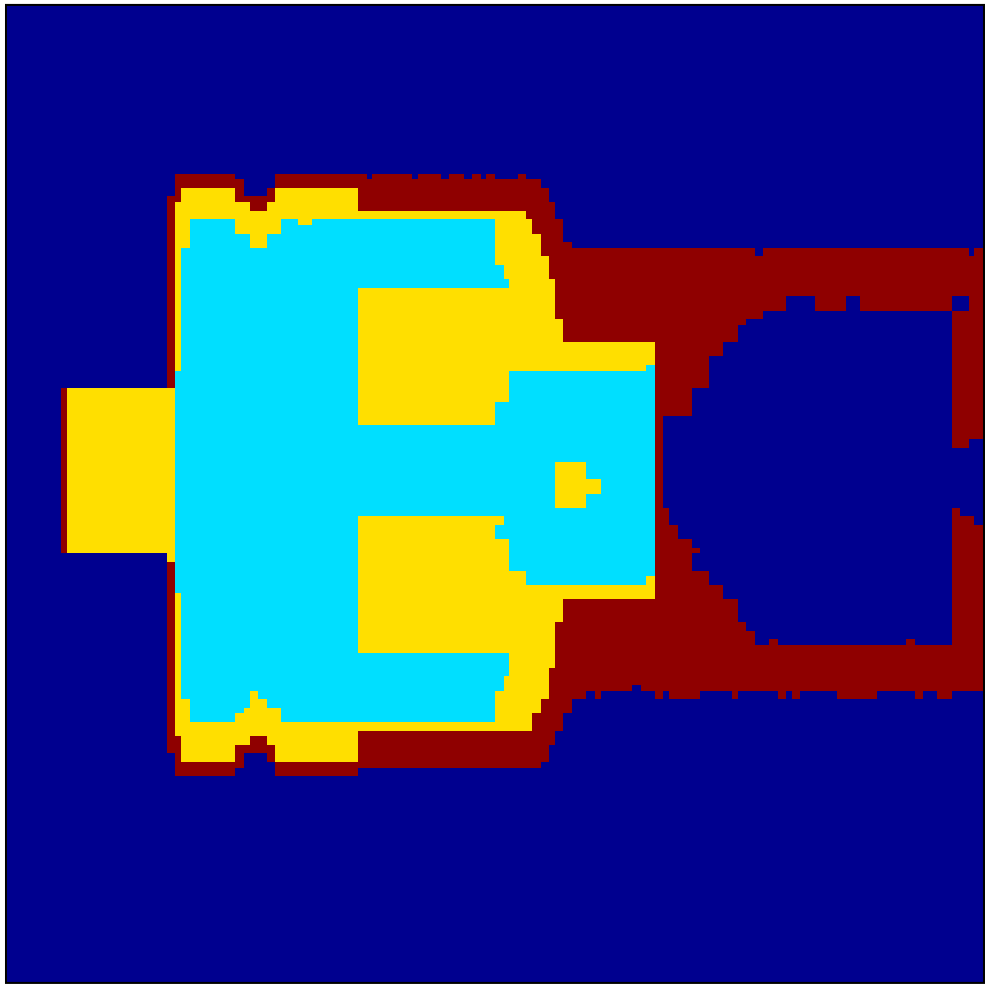
Minimum quadrant size 8x8

STUDENTS-HUB.com

Uploaded By: anonymous

Segmentation by Region Splitting and Merging





STUDENTS-HUB.com

Uploaded By: anonymous

Outline

- Element of Image Analysis
- Applications
- Classical Segmentation Techniques
  - Edge-based
  - Thresholding-based
  - Region Based
  - **Clustering Based**
  - Graph Based
  - Others
- Deep Learning Techniques
  - Semantic Segmentation
  - Instance Segmentation

STUDENTS-HUB.com

Uploaded By: anonymous

What is Clustering?

- Organizing data into classes such that:
  - High intra-class similarity
  - Low inter-class similarity
- Finding the class labels and the number of classes directly from the data
- What is similarity ?
  - **Cluster by features:**
    - **Color**
    - **Intensity**
    - **Location**
    - **Texture**
    - **….**

STUDENTS-HUB.com Uploaded By: anonymous

Feature Space

- Depending on what we choose as the feature space, we can group pixels in different ways.
- Grouping pixels based on intensity similarity

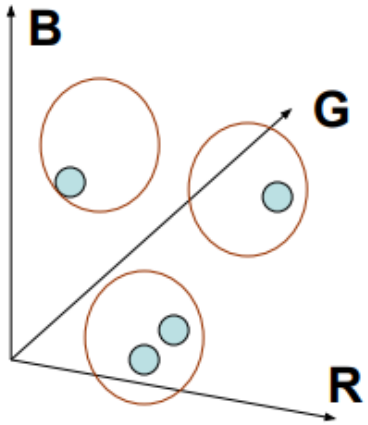
- Feature space: intensity value (1D)

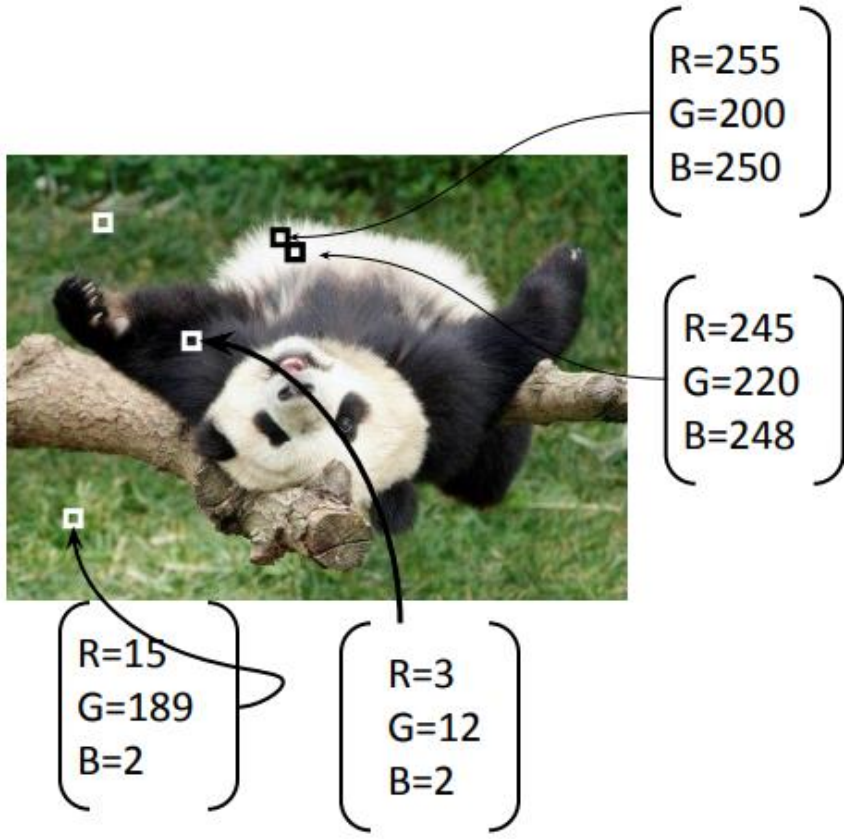
STUDENTS-HUB.com

Uploaded By: anonymous

Feature Space

- Depending on what we choose as the feature space, we can group pixels in different ways.
- Grouping pixels based on color similarity



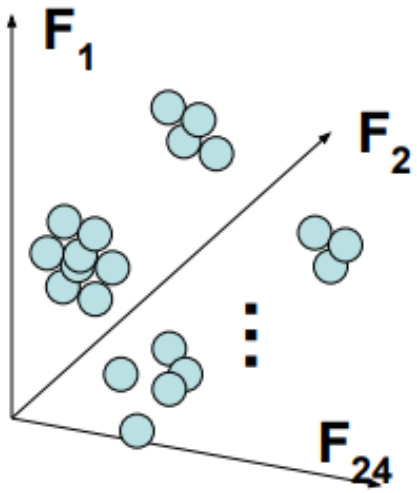


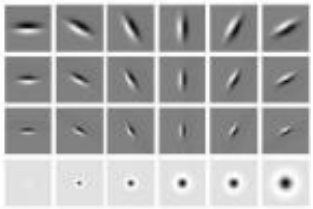
- Feature space: color value (3D)

STUDENTS-HUB.com
Uploaded By: anonymous

Feature Space

- Depending on what we choose as the feature space, we can group pixels in different ways.
- Grouping pixels based on texture similarity





Filter bank of 24 filters

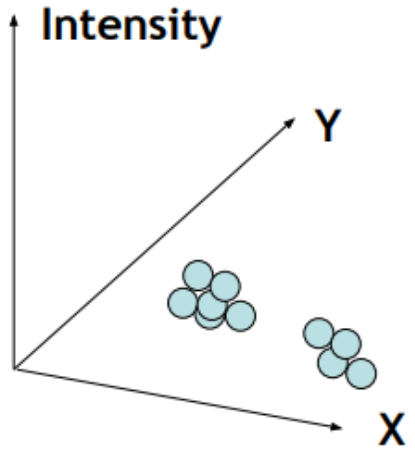
- Feature space: filter bank responses (e.g., 24D)

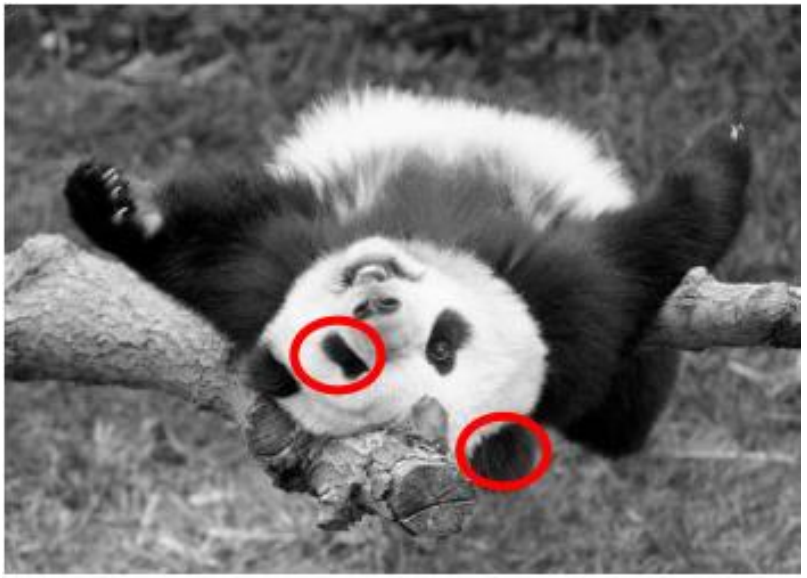
STUDENTS-HUB.com

Uploaded By: anonymous

Feature Space

- Depending on what we choose as the feature space, we can group pixels in different ways.
- Grouping pixels based on intensity+position similarity



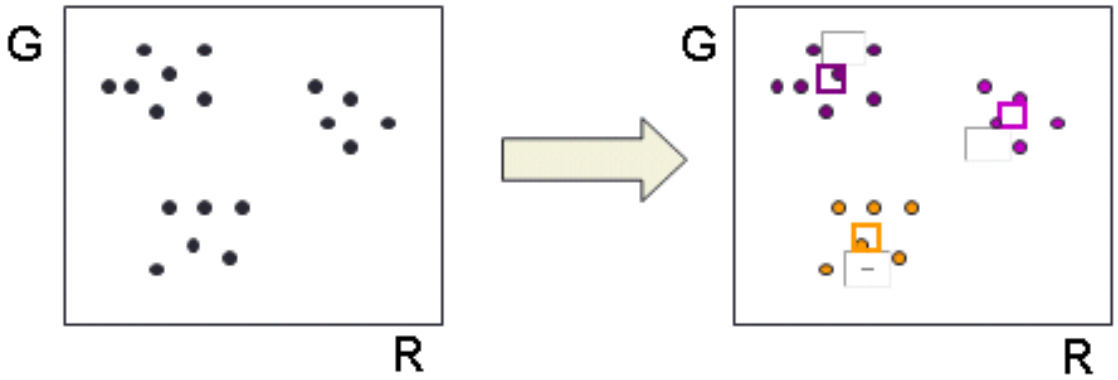


- Way to encode both similarity and proximity.

STUDENTS-HUB.com
Uploaded By: anonymous

Clustering

- How to choose the representative colors?
  - This is a clustering problem!

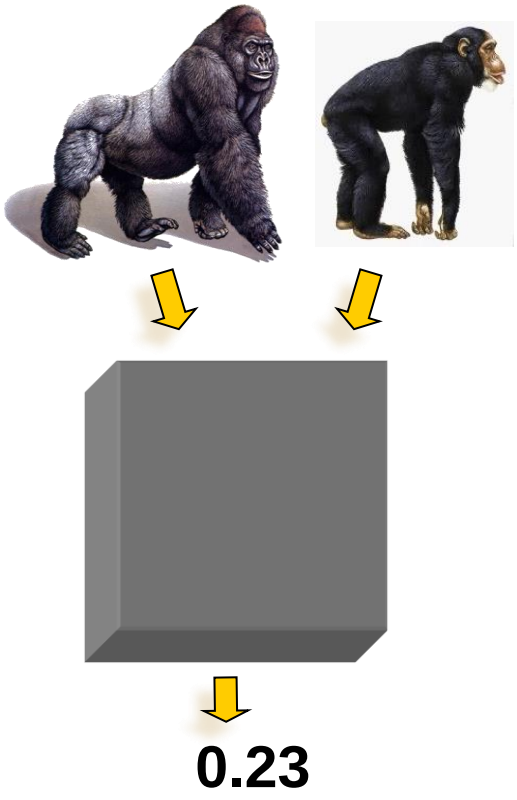


- Objective
  - Each point should be as close as possible to a cluster center
    - Minimize sum squared distance

$$\sum_{\text{clusters } i} \quad \sum_{\text{points p in cluster } i} \|p - c_i\|^2$$

STUDENTS-HUB.com Uploaded By: anonymous

Distance metrics



Uploaded By: anonymous

K-means

- K-means clustering based on intensity or color is essentially vector quantization of the image attributes
  - Clusters don't have to be spatially coherent
- Given a K, find a partition of K clusters to optimize the chosen partitioning criterion (cost function)
  - global optimum: exhaustively search all partitions
- The K-means algorithm: a heuristic method
  - K-means algorithm (MacQueen'67): each cluster is represented by the centre of the cluster and the algorithm converges to stable centroids of clusters.
  - K-means algorithm is the simplest partitioning method for clustering analysis and widely used in data mining applications.

STUDENTS-HUB.com Uploaded By: anonymous

K-Means Clustering Algorithm

- Given the cluster number *K*, the *K-means* algorithm is carried out in three steps after initialization:

Initialisation: set seed points (randomly)

1) Assign each object to the cluster of the nearest seed point measured with a specific distance metric
2) Compute new seed points as the centroids of the clusters of the current partition (the centroid is the centre, i.e., *mean point*, of the cluster)
3) Go back to Step 1), stop when no more new assignment (i.e., membership in each cluster no longer changes)

STUDENTS-HUB.com Uploaded By: anonymous

Stopping/convergence criterion

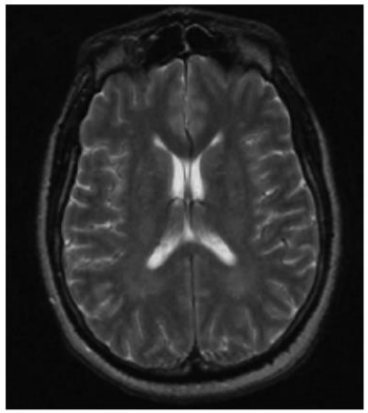
1. No (or minimum) re-assignments of data points to different clusters,
2. No (or minimum) change of centroids, or
3. Minimum decrease in the **sum of squared error** (SSE),

$$SSE = \sum_{j=1}^{k} \sum_{\mathbf{x} \in C_j} dist(\mathbf{x}, \mathbf{m}_j)^2$$

C_i is the jth cluster, $\mathbf{m}_j$ is the centroid of cluster C_j (the mean vector of all the data points in C_j), and $dist(\mathbf{x}, \mathbf{m}_j)$ is the distance between data point $\mathbf{x}$ and centroid $\mathbf{m}_j$.

STUDENTS-HUB.com Uploaded By: anonymous

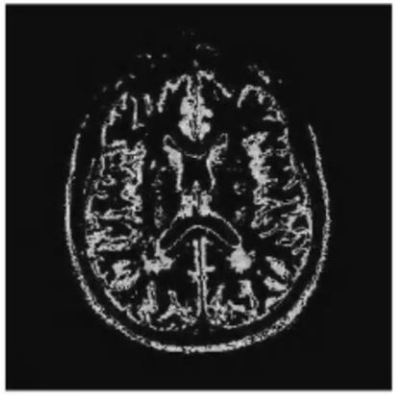
K-means Image Segmentation - Example



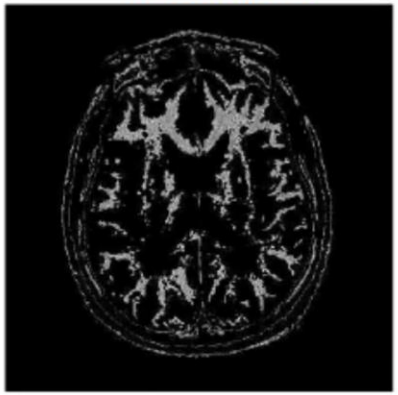
k=9 All Segmentations Shown

k=9 Cluster 1

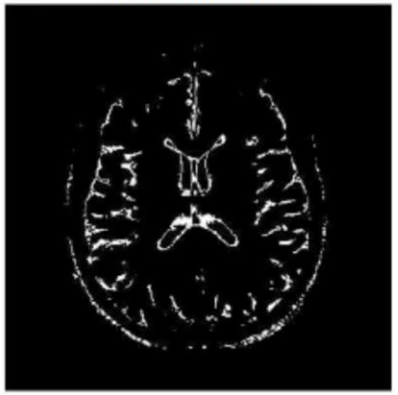
k=9 Cluster 4



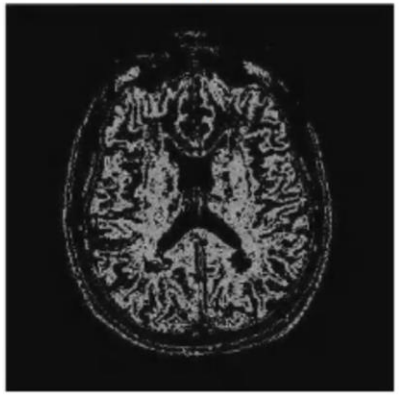
k=9 Cluster 5



k=9 Cluster 6



k=9 Cluster 9



Uploaded By: anonymous

K-means clustering using intensity or color

Image

Clusters on intensity

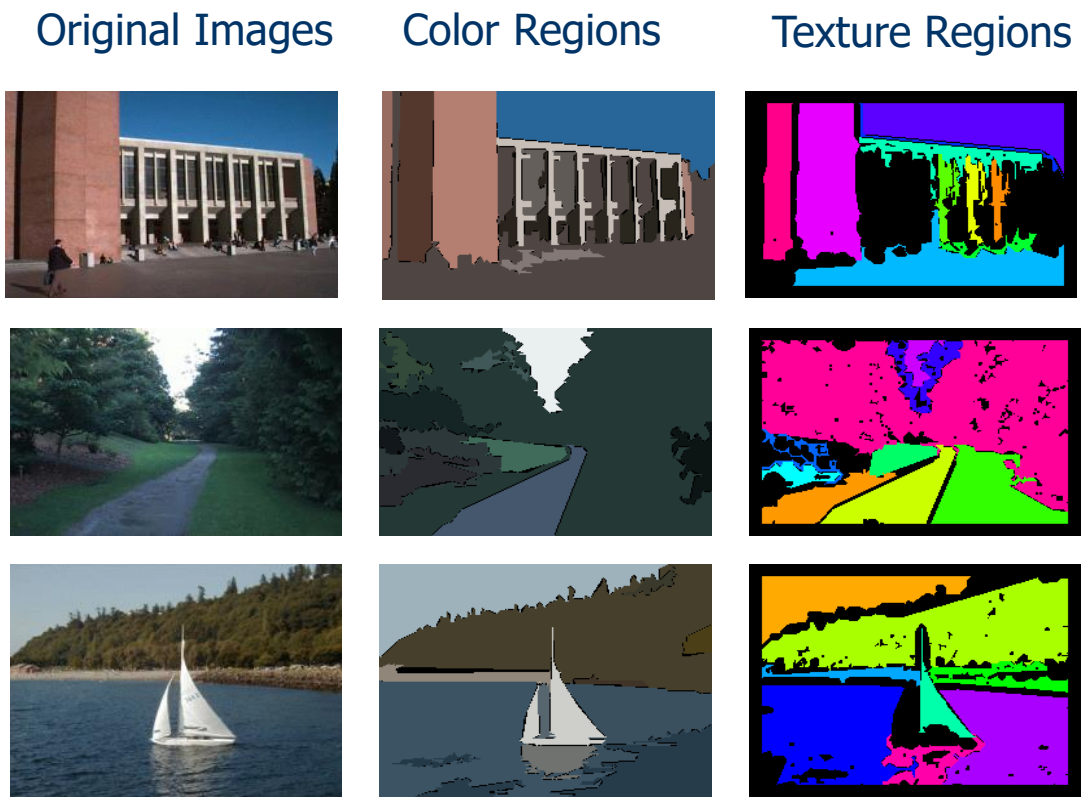
Clusters on color

STUDENTS-HUB.com

Uploaded By: anonymous

Clustering-based segmentation

- Clustering can also be used with other features (e.g., texture) in addition to color.



STUDENTS-HUB.com
Uploaded By: anonymous

Other Clustering Approaches

- Mean Shift
- Hierarchical approach
  - Agglomerative, Diana, Agnes, BIRCH, ROCK, ……
- Density-based approach
  - DBSACN, OPTICS, DenClue, ……
- Model-based approach
  - Gaussian Mixture Model (GMM), COBWEB, ……
- Grid-based Approach
  - STING, CLIQUE, Wave-Cluster…
- Spectral clustering approach
  - Normalised-Cuts…

STUDENTS-HUB.com Uploaded By: anonymous

Outline

- Element of Image Analysis
- Applications
- Classical Segmentation Techniques
  - Edge-based
  - Thresholding-based
  - Region Based
  - Clustering Based
  - **Graph Based**
  - Others
- Deep Learning Techniques
  - Semantic Segmentation
  - Instance Segmentation

STUDENTS-HUB.com

Uploaded By: anonymous

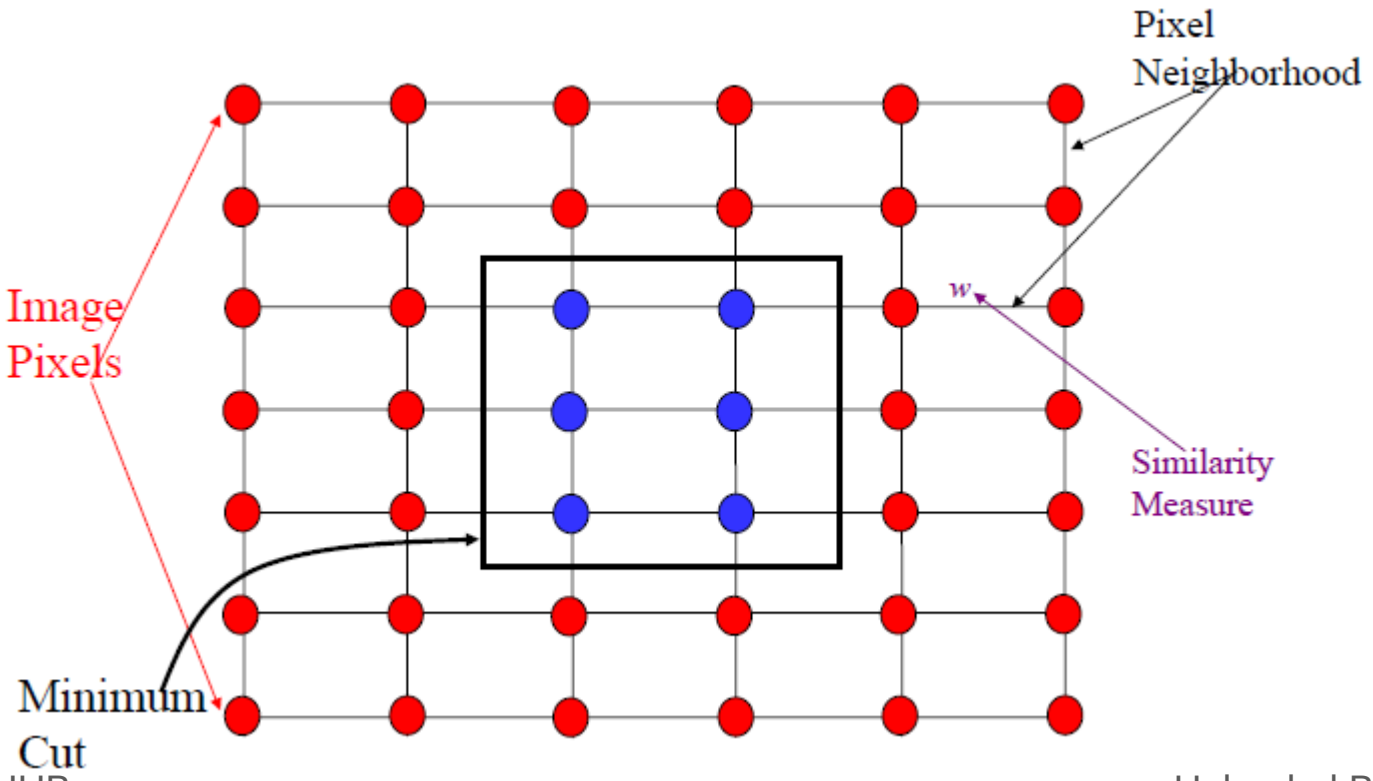
Graph Cut Segmentation

- Graph representation of an Image
  - Set of points of the feature space represented as a weighted, undirected graph, G = (V, E).
  - The points of the feature space are the nodes of the graph.
  - Edge between every pair of nodes.
  - Weight on each edge, *w(i, j)*, is a function of the similarity between the nodes i and j.
  - Partition the set of vertices into disjoint sets where similarity within the sets is high and across the sets is low.
  - Therefore, Segmentation is equivalent to Graph partition.

STUDENTS-HUB.com Uploaded By: anonymous

Images as graphs

- Fully-connected graph
  - node for every pixel
  - link between every pair of pixels, **p**,**q**
  - similarity w_{ij} for each link



STUDENTS-HUB.com
Uploaded By: anonymous

Measuring Affinity

Distance

$$aff(x,y) = \exp\left\{-\left(1/2\sigma_d^2\right)\left(\|x-y\|^2\right)\right\}$$

Intensity

$$aff(x,y) = \exp\left\{-\left(1/2\sigma_i^2\right)\left(\|I(x)-I(y)\|^2\right)\right\}$$

Color

$$aff(x,y) = \exp\left\{-\left(1/2\sigma_t^2\right)\left(\|c(x)-c(y)\|^2\right)\right\}$$

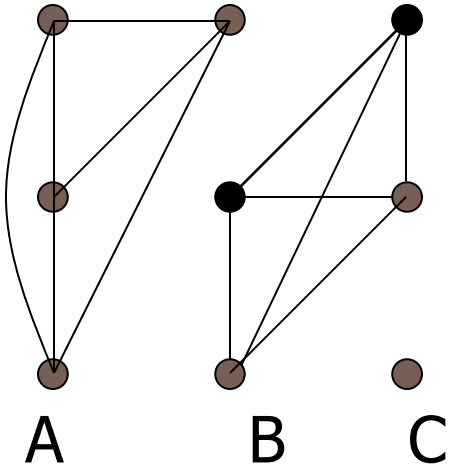
Texture

$$aff(x,y) = \exp\left\{-\frac{1}{2\sigma_d^2}\underbrace{\|f(x)-f(y)\|^2}\right\}$$

STUDENTS-HUB.com

Uploaded By: anonymous

Segmentation by graph partitioning



- Break Graph into Segments
  - Delete links that cross between segments
  - Easiest to break links that have low affinity
    - Similar pixels should be in the same segments
    - Dissimilar pixels should be in different segments

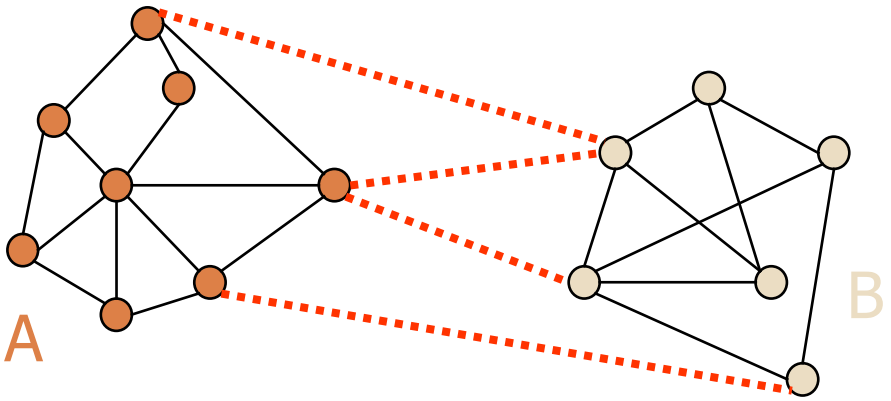
STUDENTS-HUB.com Uploaded By: anonymous

Graph cut

- Set of edges whose removal makes a graph disconnected
- Cost of a cut: sum of weights of cut edges:

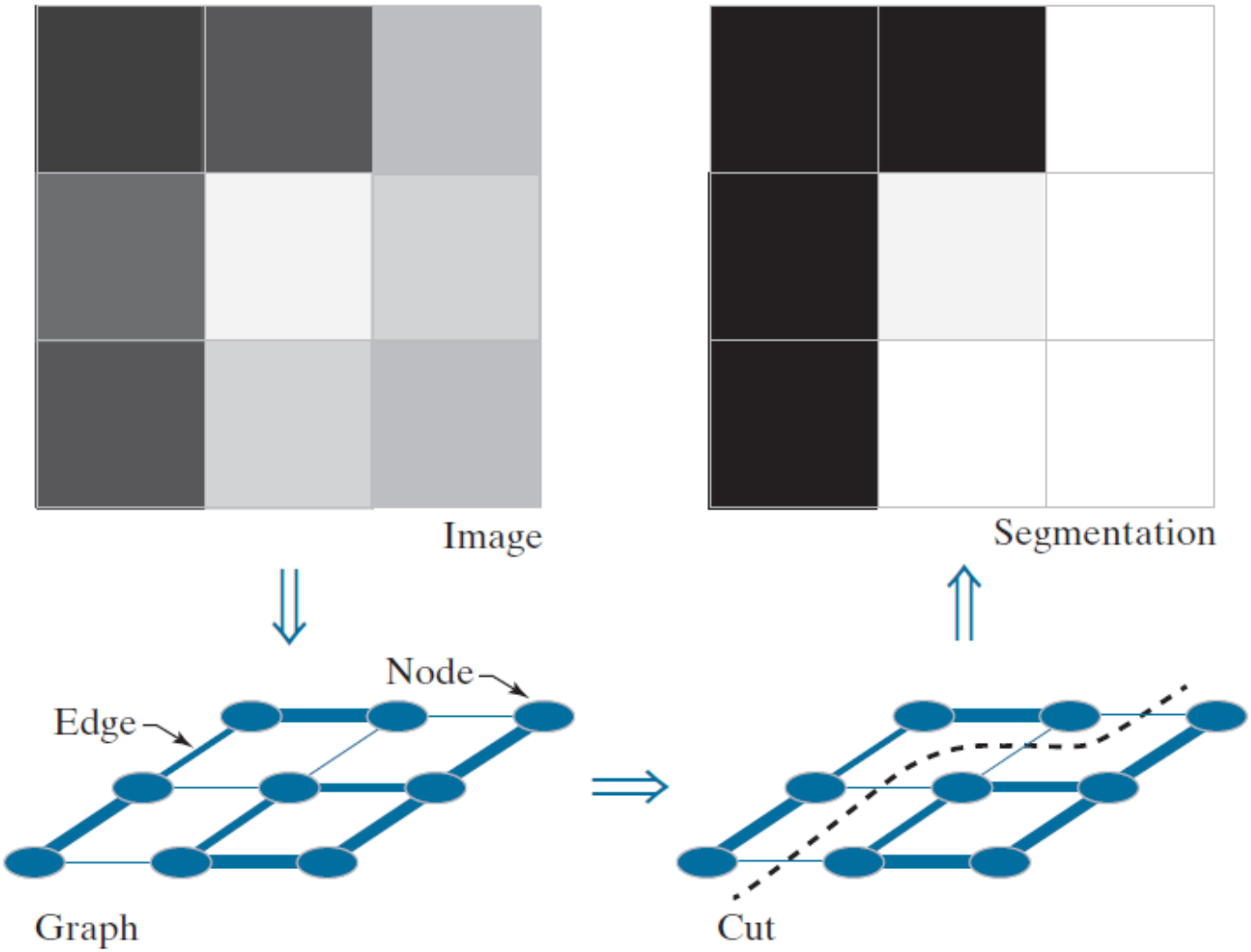
$$cut(A,B) = \sum_{p \in A, q \in B} w_{p,q}$$

- A graph cut gives us a segmentation
- The main problems to be solved during graph-based image segmentation is how to choose edges to be removed in order to divide graph into pieces.
- What is a “good” graph cut and how do we find one?



STUDENTS-HUB.com
Uploaded By: anonymous

Graph Cut - Example



STUDENTS-HUB.com
Uploaded By: anonymous

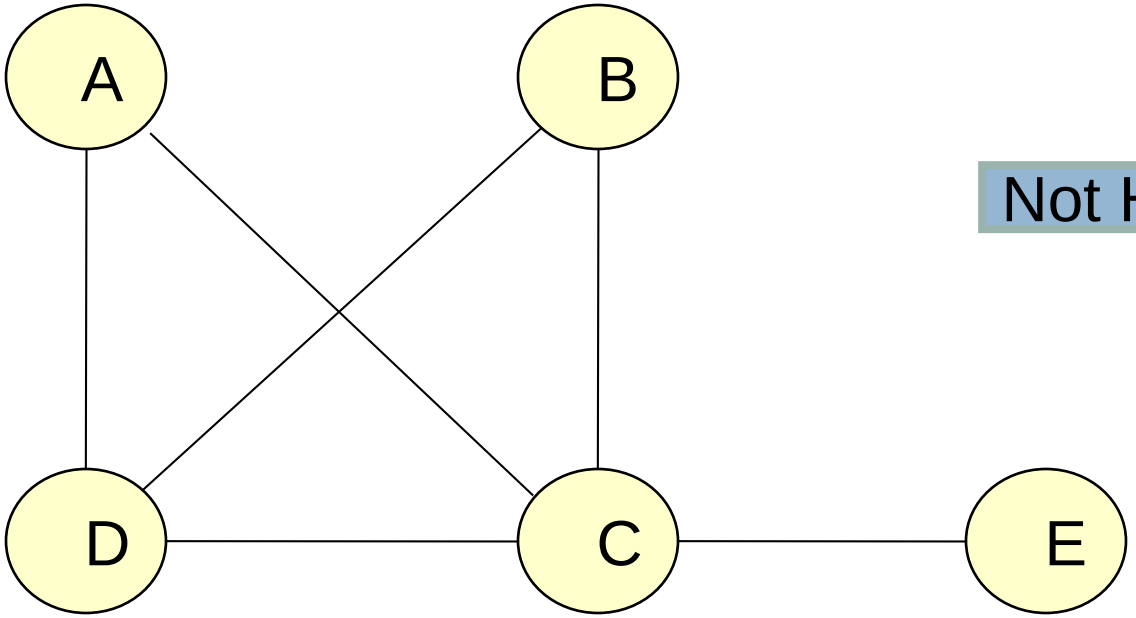
Segmentation by Min Cut method

- By Min Cut method, the graph is partitioned into clusters.
- Each cluster is considered as an image segment.
- Min Cut method uses the HCS (Highly Connected Subgraphs) Algorithm to find the clusters.
- Edge Connectivity is the minimum number of edges whose removal results in a disconnected graph. It is denoted by k(G).
- For a graph with vertices n > 1 to be highly connected if its edge-connectivity k(G) > n/2.
- A highly connected subgraph (HCS) is an induced subgraph H in G such that H is highly connected.

STUDENTS-HUB.com Uploaded By: anonymous

Segmentation by Min Cut method

Example:



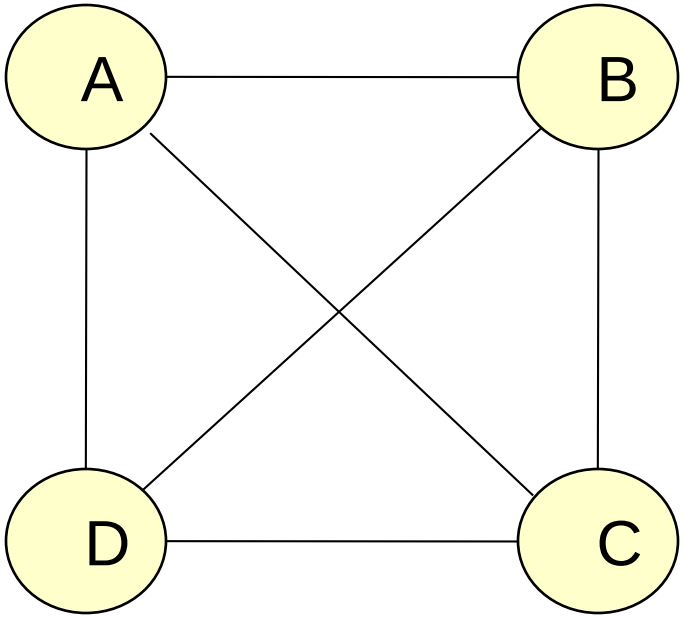
Not HCS!

No. of nodes = 5 Edge Connectivity = 1

STUDENTS-HUB.com Uploaded By: anonymous

Segmentation by Min Cut method

Example continued:



HCS!

No. of nodes = 4 Edge Connectivity = 3

STUDENTS-HUB.com Uploaded By: anonymous

HCS Algorithm - Min Cut method


```
HCS(G(V,E))
begin
    (H, H',C) ← MINCUT(G)
    if G is highly connected
            then return (G)
    else
            HCS(H)
            HCS(H')
    end if
end
```

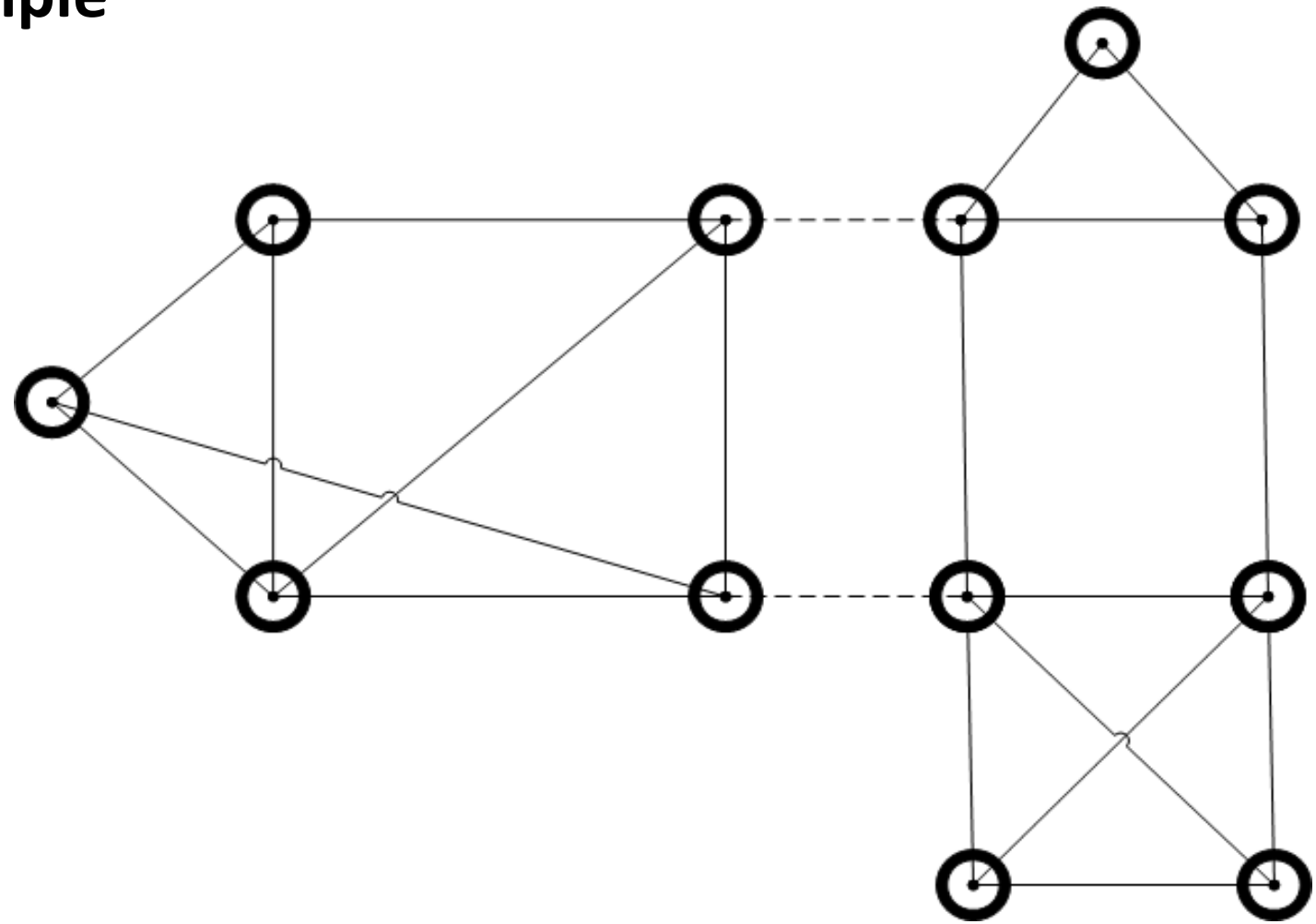
- The procedure MINCUT(G) returns H, H' and C where C is the minimum cut which separates G into the subgraphs H and H'.
- Procedure HCS returns a graph in case it identifies it as a cluster.
- Single vertices are not considered clusters and are grouped into singletons set S.

STUDENTS-HUB.com

Uploaded By: anonymous

HCS Algorithm - Min Cut method

Example

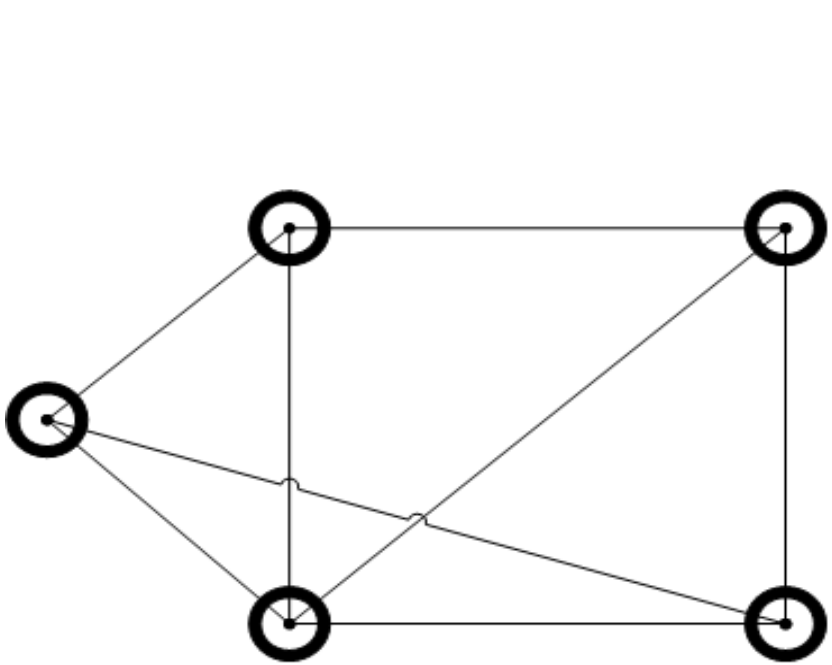


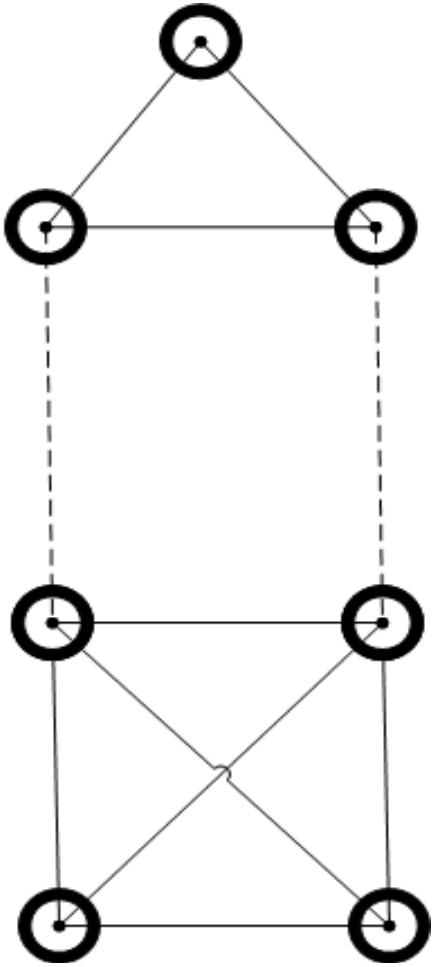
STUDENTS-HUB.com

Uploaded By: anonymous

HCS Algorithm - Min Cut method

Example Continued





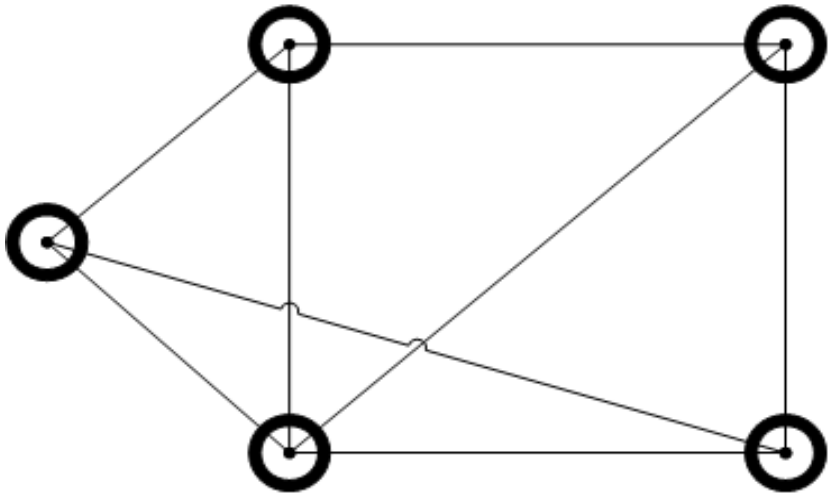
STUDENTS-HUB.com

Uploaded By: anonymous

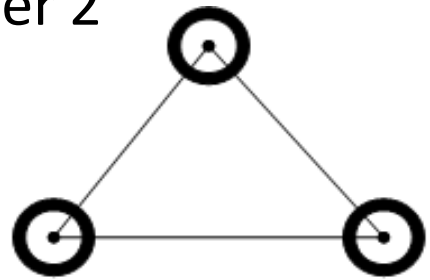
HCS Algorithm - Min Cut method

Example Continued

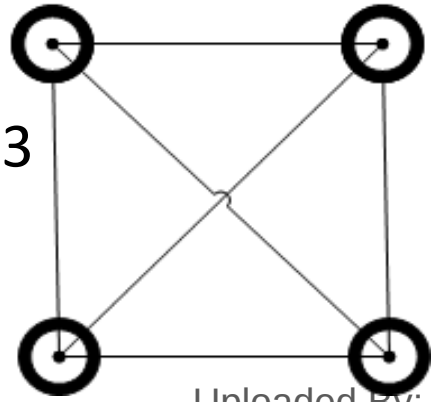
Cluster 1



Cluster 2



Cluster 3



STUDENTS-HUB.com

Uploaded By: anonymous

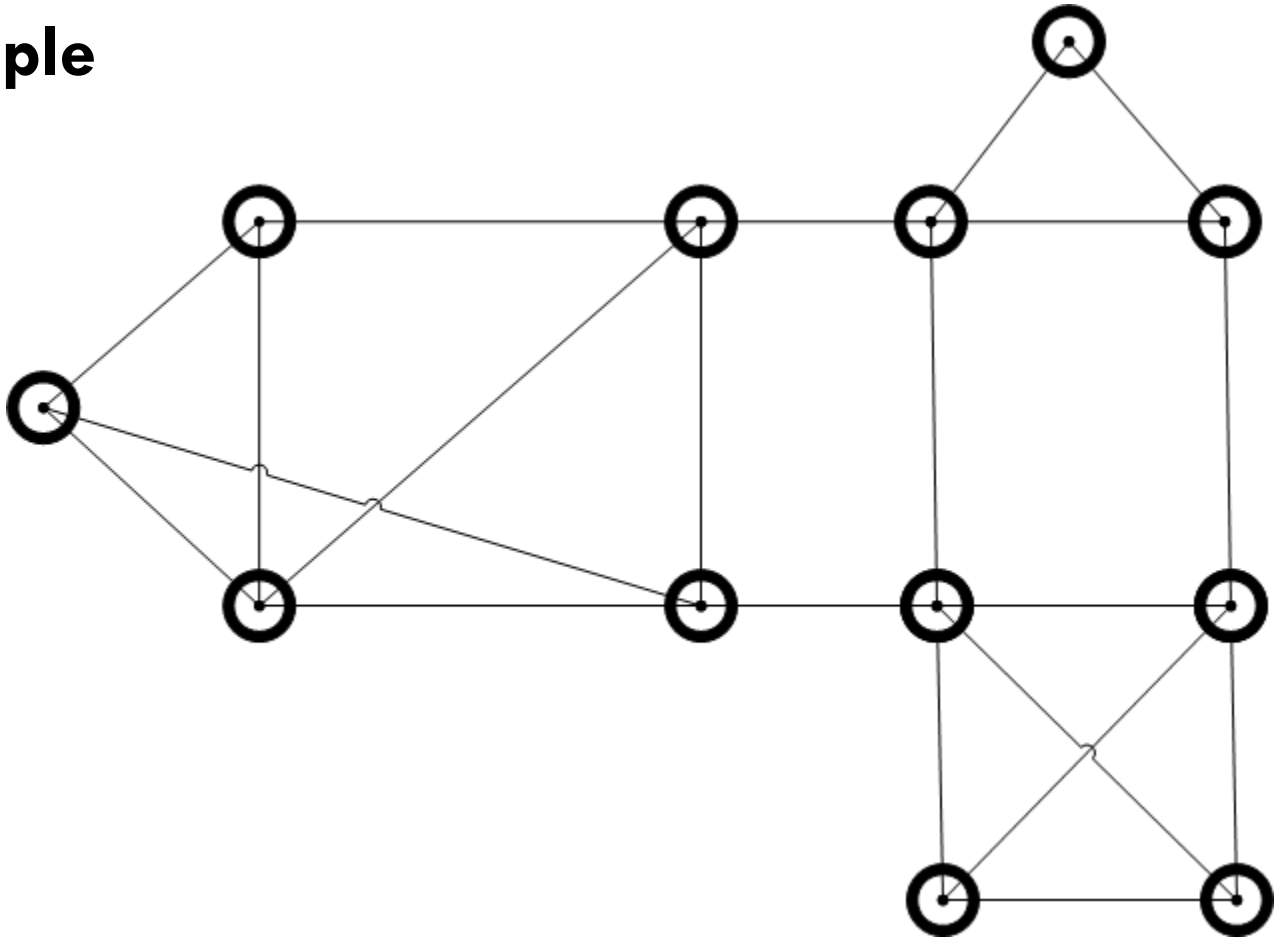
Min Cut method

- We can do segmentation by finding the *minimum cut* in a graph
  - A **minimum cut** of a graph is a cut whose cutset has the smallest number of elements (unweighted case) or smallest sum of weights possible.
  - Efficient algorithms exist for doing this
- Drawback:
  - Weight of cut proportional to number of edges in the cut
  - Minimum cut tends to cut off very small, isolated components
  - To overcome this we use normalized-cut for image segmentation.

STUDENTS-HUB.com Uploaded By: anonymous

HCS Algorithm – Main Problem

Example

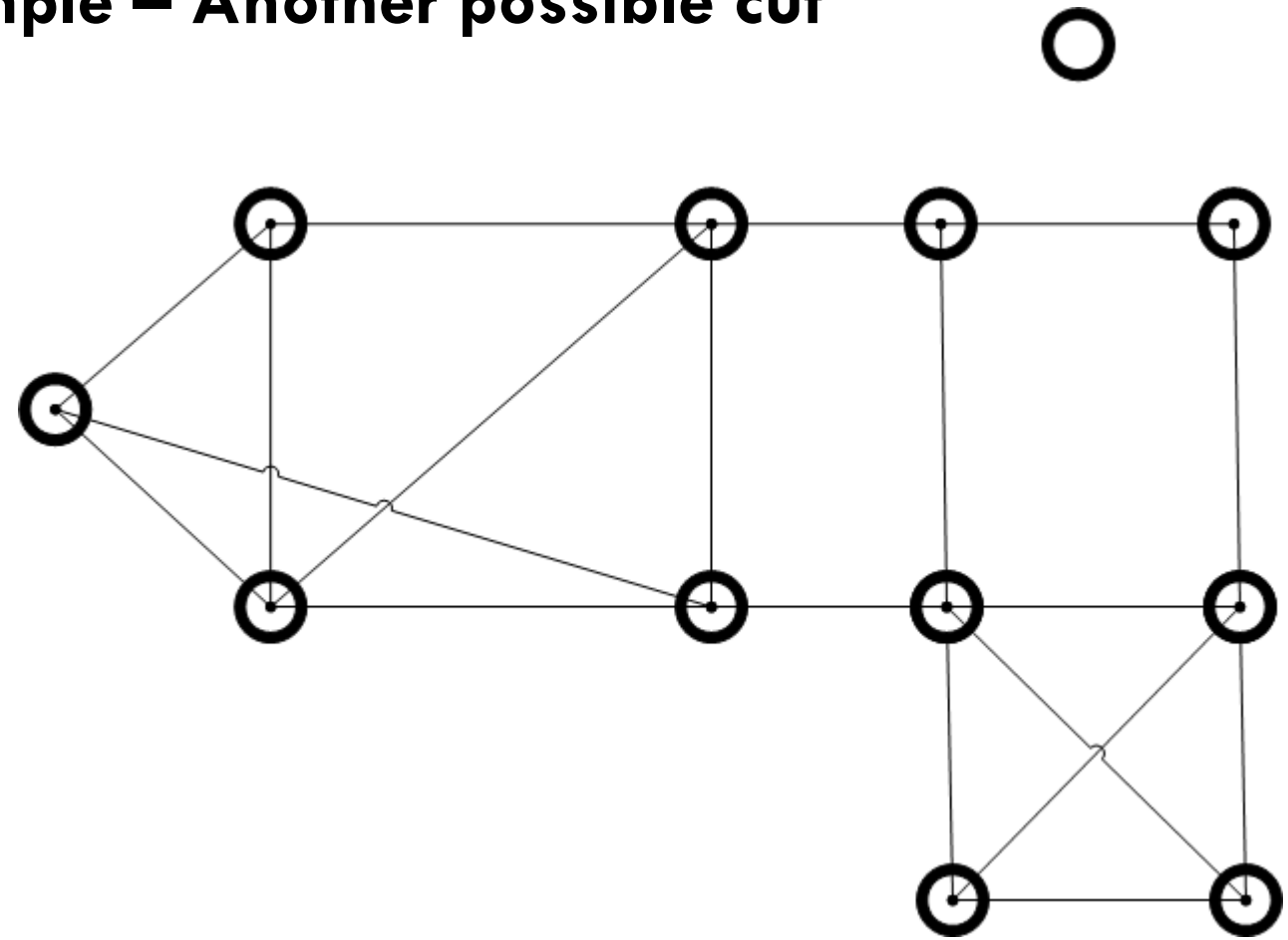


STUDENTS-HUB.com

Uploaded By: anonymous

HCS Algorithm – Main Problem

Example – Another possible cut



STUDENTS-HUB.com

Uploaded By: anonymous

Normalized Cut

- Instead of looking at the value of total edge weight connecting the two partitions, Normalized measure computes the cut cost as a fraction of the total edge connections to all the nodes in the graph.

$$Ncut(A,B) = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)},$$

Where $assoc(A,V) = \sum_{u \in A, t \in V} w(u,t)$

is the total connection from nodes in A to all nodes in the graph

- By using *Ncut*(*A*,*B*) instead of *cut*(*A*,*B*), the cut that partitions isolated points will no longer have small values
  - if *A* is a single node, *cut*(*A*,*B*) and *assoc*(*A*,*V*) will have the same value.
  - Thus, independently of how small *cut*(*A*,*B*) is, *Ncut*(*A*,*B*) will always be greater than or equal to 1, thus providing normalization for "pathological" cases such as this.

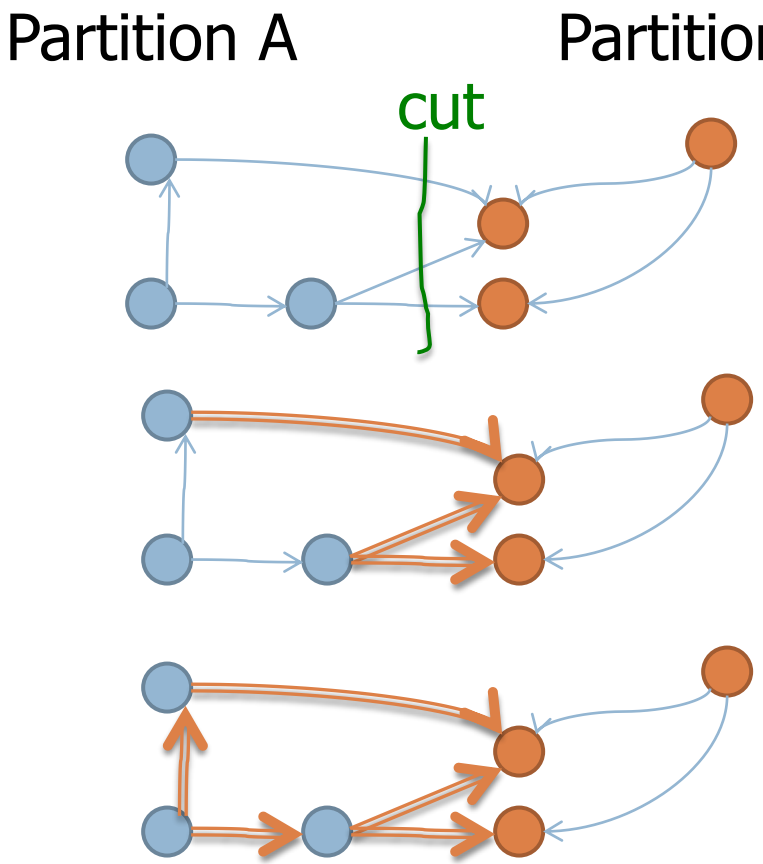
STUDENTS-HUB.com Uploaded By: anonymous

Normalized Cut

$$Ncut(A,B)$$

$$cut(A,B)$$

$$\overline{assoc(A,V)}$$



STUDENTS-HUB.com
Uploaded By: anonymous

Normalized Cut

- We can define a measure for total *normalized association* within graph partitions as

$$Nassoc(A,B) = \frac{assoc(A,A)}{assoc(A,V)} + \frac{assoc(B,B)}{assoc(B,V)}$$

where *assoc*(*A*,*A*) and *assoc*(*B*,*B*) are the total weights connecting the nodes within *A* and within *B*, respectively.

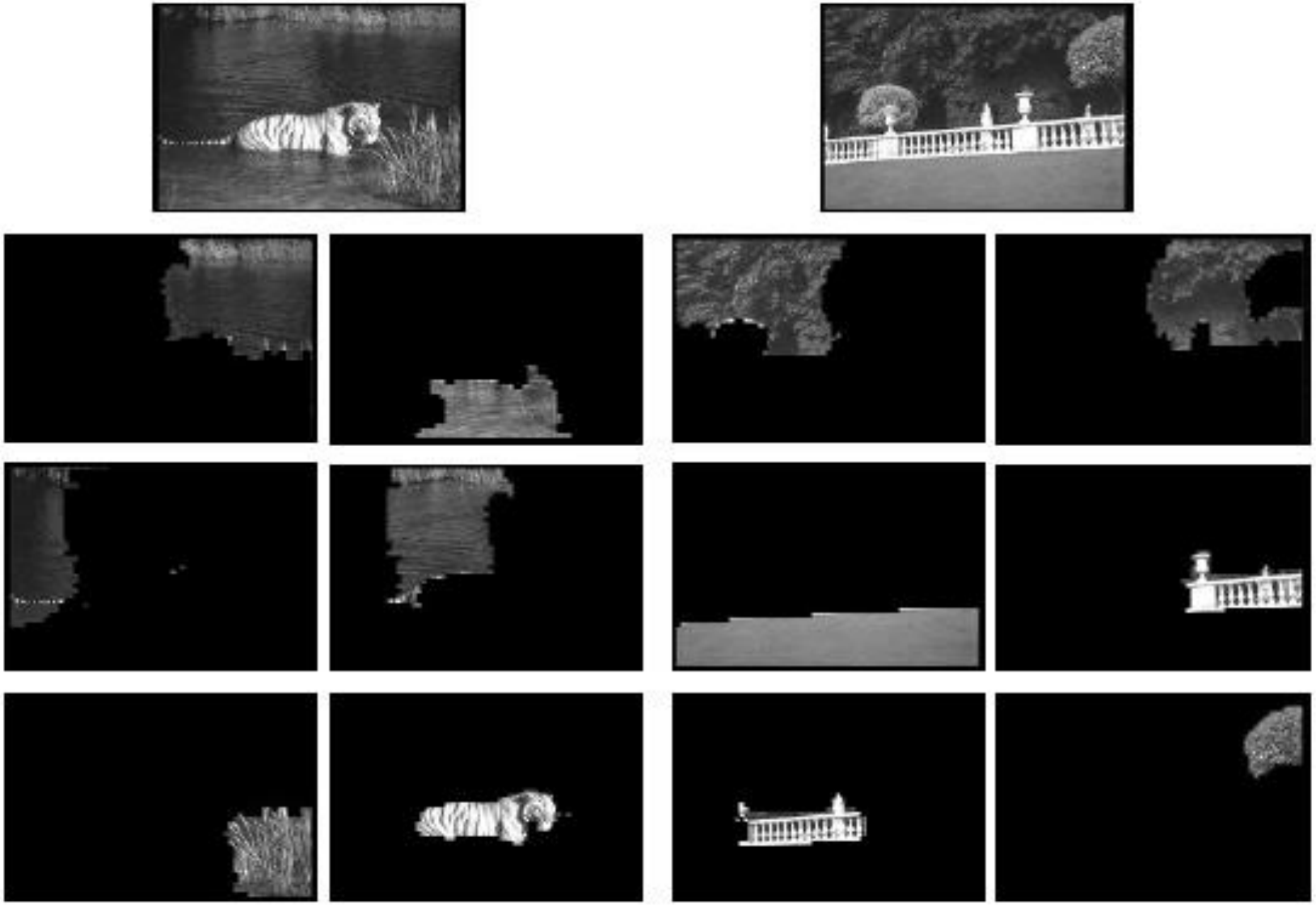
$$Ncut(A,B) = 2 - Nassoc(A,B)$$

which implies that minimizing *Ncut*(*A*,*B*) simultaneously maximizes *Nassoc*(*A*,*B*).

- Therefore, image segmentation using graph cuts is now based on finding a partition that minimizes *Ncut*(*A*,*B*). Unfortunately, minimizing this quantity exactly is an NP-complete computational task

STUDENTS-HUB.com Uploaded By: anonymous

Example result



STUDENTS-HUB.com Uploaded By: anonymous

Other Graph Cut Methods

- Average Cut
- Max Flow Cut
- Some Notes:
  - The performances for the different cut measures are not significantly different from each other
  - The normalized cut based partition took significantly larger time to compute than the average and the min cut

STUDENTS-HUB.com Uploaded By: anonymous

Other Classical Segmentation Methods

- Active contour
- Level Set Segmentation
- Deformable templates segmentation
- Segmentation As an Energy minimization problem
  - MRF
- Segmentation by morphological watersheds
- …..

STUDENTS-HUB.com Uploaded By: anonymous

Outline

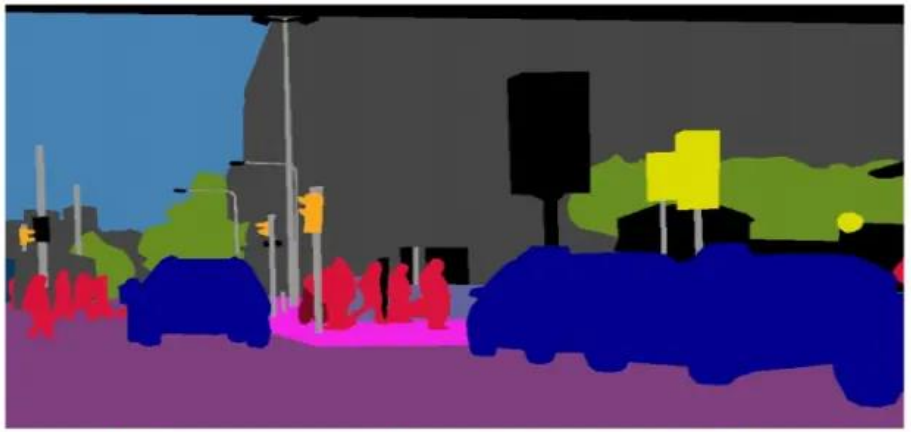
- Element of Image Analysis
- Applications
- Classical Segmentation Techniques
  - Edge-based
  - Thresholding-based
  - Region Based
  - Clustering Based
  - Graph Based
  - Others
- Deep Learning Techniques
  - Semantic Segmentation
  - Instance Segmentation
  - Panoptic Segemntation

STUDENTS-HUB.com

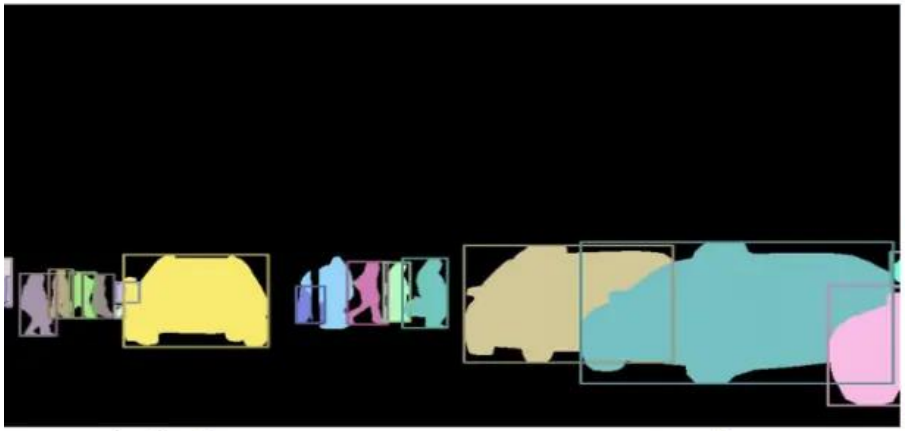
Uploaded By: anonymous

Deep Learning Segmentation Methods

(a) image



(b) semantic segmentation



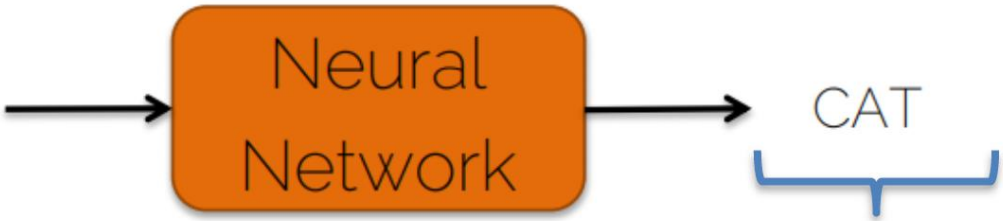
(c) instance segmentation

(d) panoptic segmentation

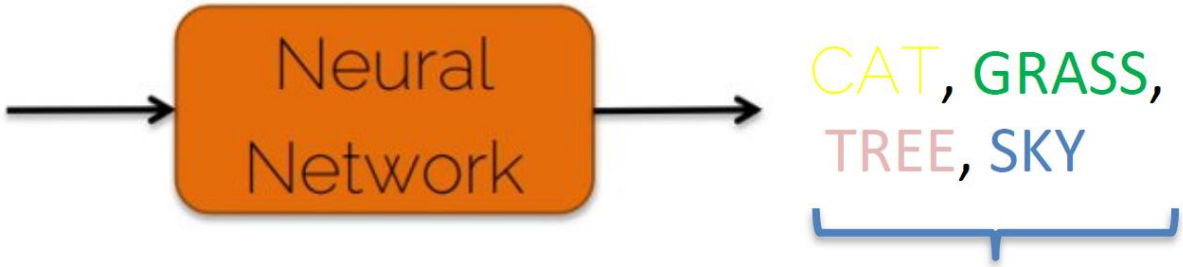
STUDENTS-HUB.com

Uploaded By: anonymous

Semantic Segmentation



Classify the main object in the image.

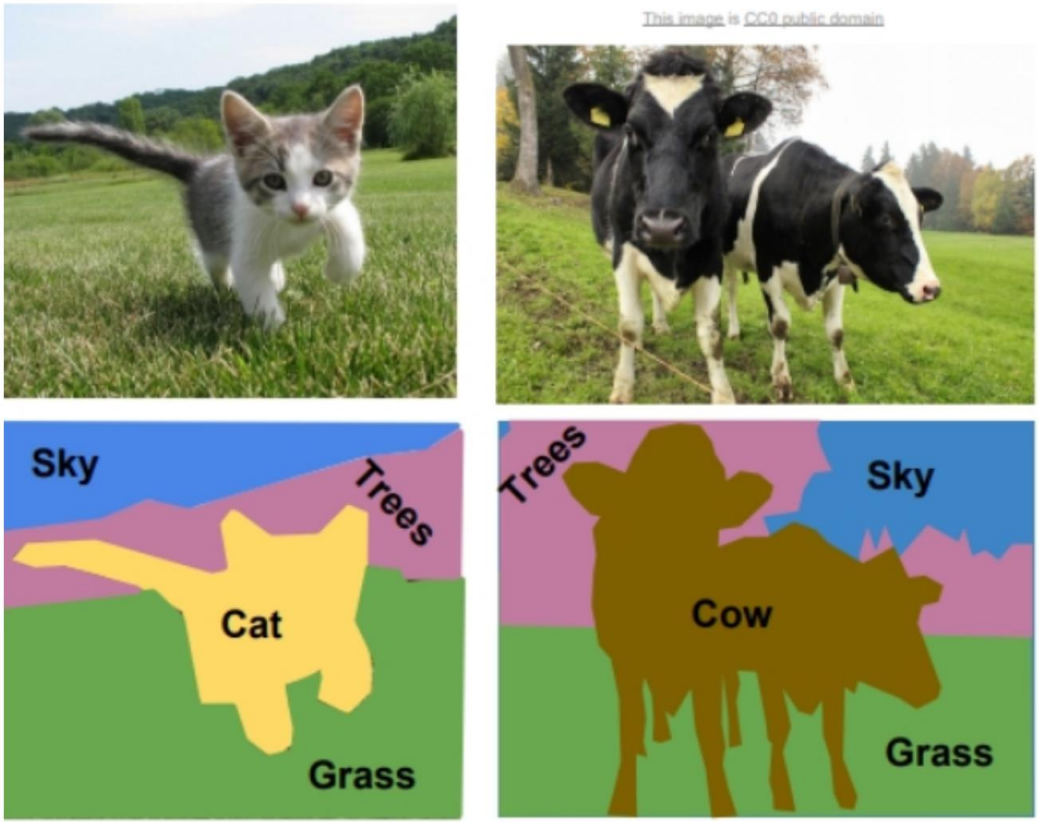


No objects, just classify each pixel.

STUDENTS-HUB.com

Uploaded By: anonymous

Semantic Segmentation



- Every pixel in the image needs to be labelled with a category label.

- Do not differentiate between the instances (see how we do not differentiate between pixels coming from different cows).

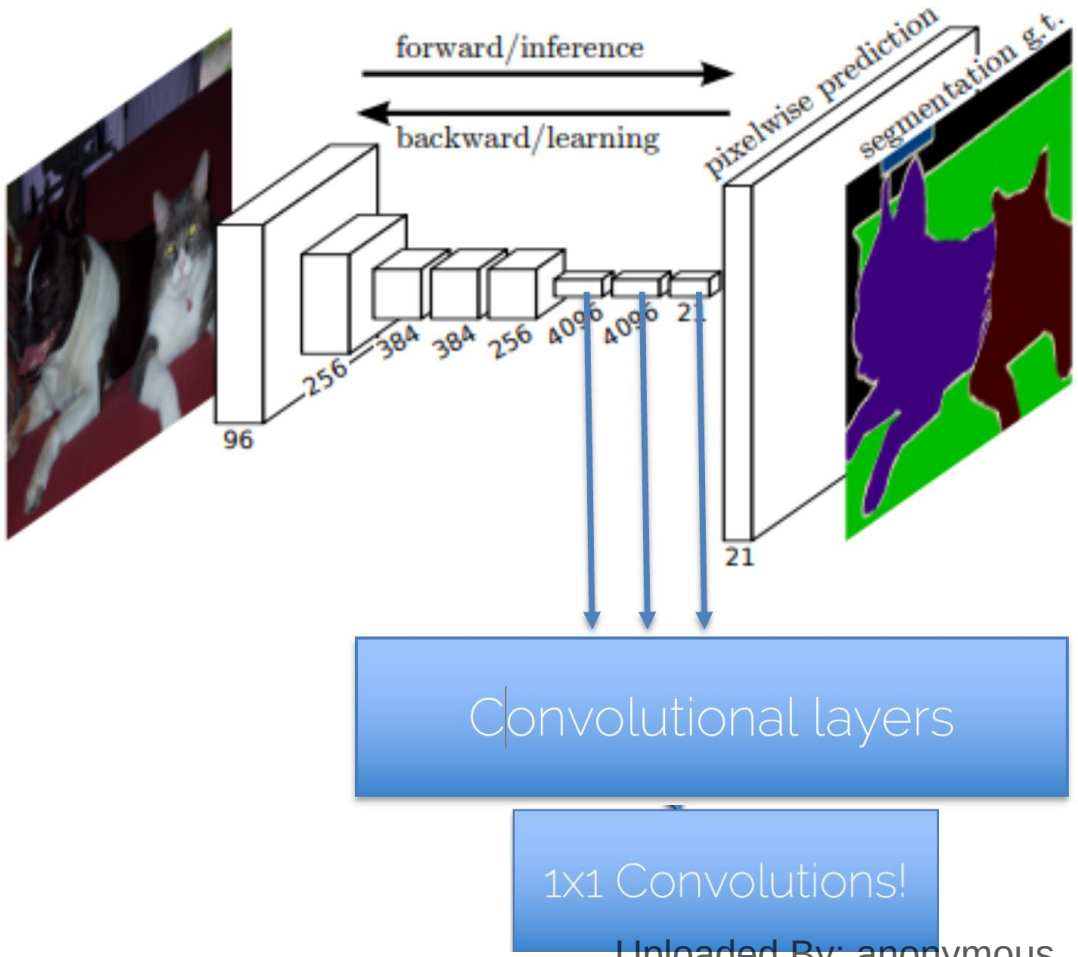
Fully Convolutional Network – Semantic Segemntation

- **Use of Convolutional Layers**: FCNs replace fully connected layers with convolutional layers, allowing them to process input images of arbitrary sizes while maintaining spatial information.
- **End-to-End Training**: FCNs are trained end-to-end to directly map input images to corresponding segmentation maps without requiring separate preprocessing, extracting the region proposals, or feature extraction steps.
- **Pixel-Wise Classification**: The network performs pixel-level classification, assigning a class label to every pixel in the image to generate dense segmentation outputs.
- **Upsampling for Dense Prediction**: Techniques like deconvolution or transpose convolution are employed to upsample the feature maps, restoring them to the original input resolution for detailed segmentation.
- **Skip Connections:** Many FCN architectures incorporate skip connections, which combine feature maps from different layers of the network. This helps to preserve fine-grained details and improve the overall accuracy of the segmentation.

STUDENTS-HUB.com Uploaded By: anonymous

Basic Fully Convolutional Networks for Semantic Segmentation

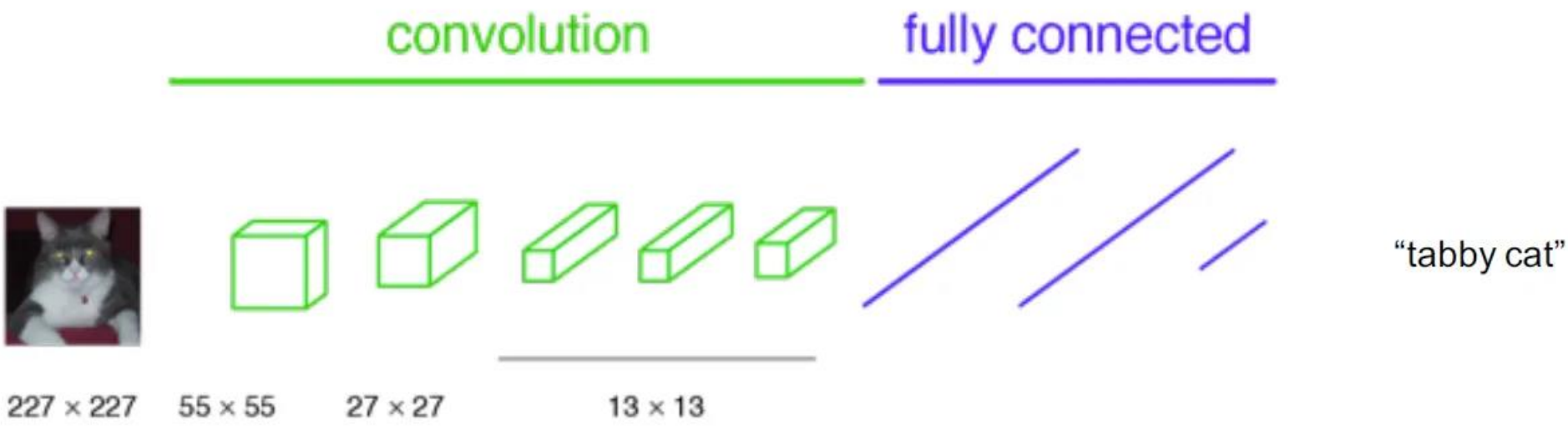
- The FCN repurposes the VGG16 network, which has 13 convolutional layers and 3 fully connected layers
- Replace FC layers with convolutional layers.
- Convert the last layer output to the original resolution.
- Do softmax-cross entropy between the pixelwise predictions and segmentaion ground truth.
- Backprop and SGD



STUDENTS-HUB.com
Uploaded By: anonymous

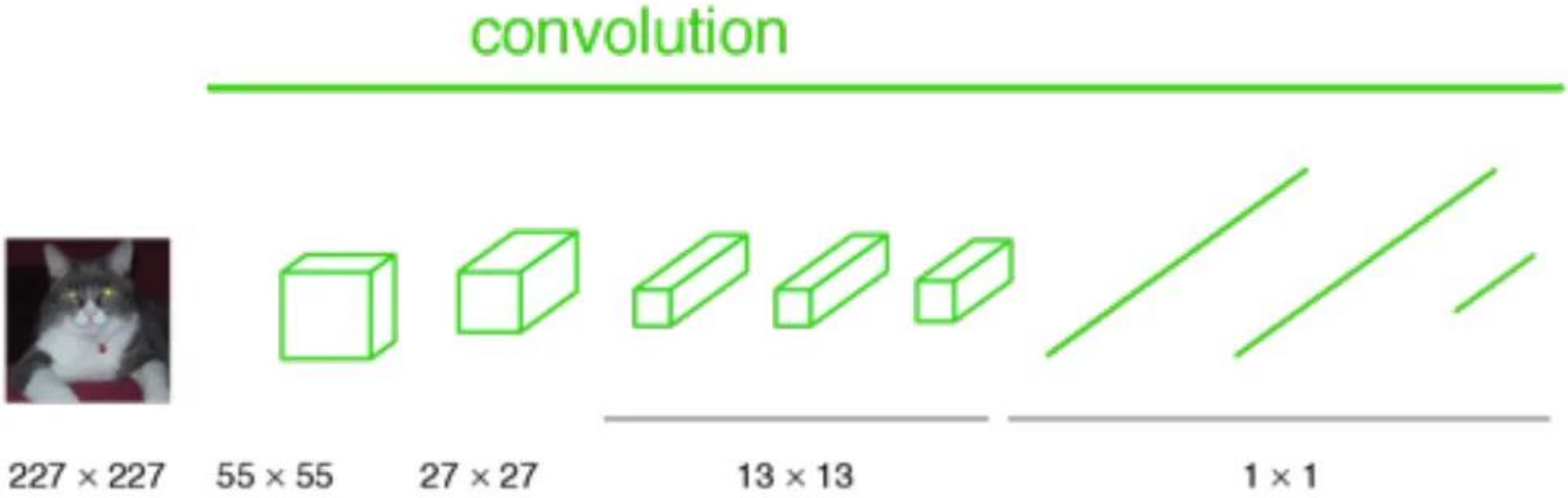
From Image Classification to Semantic Segmentation

- In classification, conventionally, an input image is downsized and goes through the convolution layers and fully connected (FC) layers, and output one predicted label for the input image



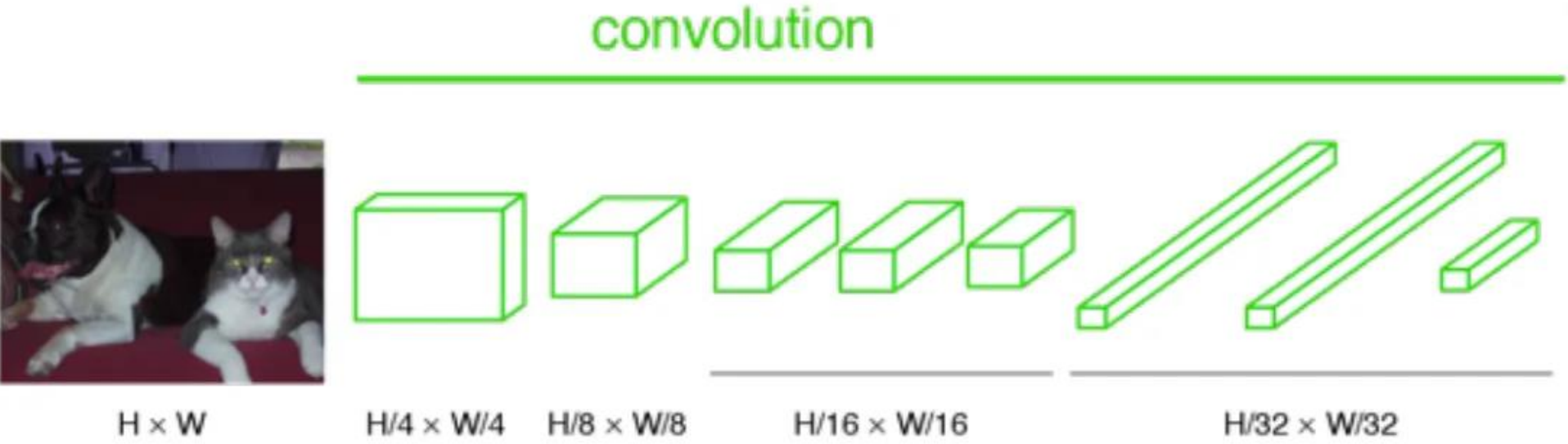
From Image Classification to Semantic Segmentation

- Imagine we turn the FC layers into 1×1 convolutional layers



From Image Classification to Semantic Segmentation

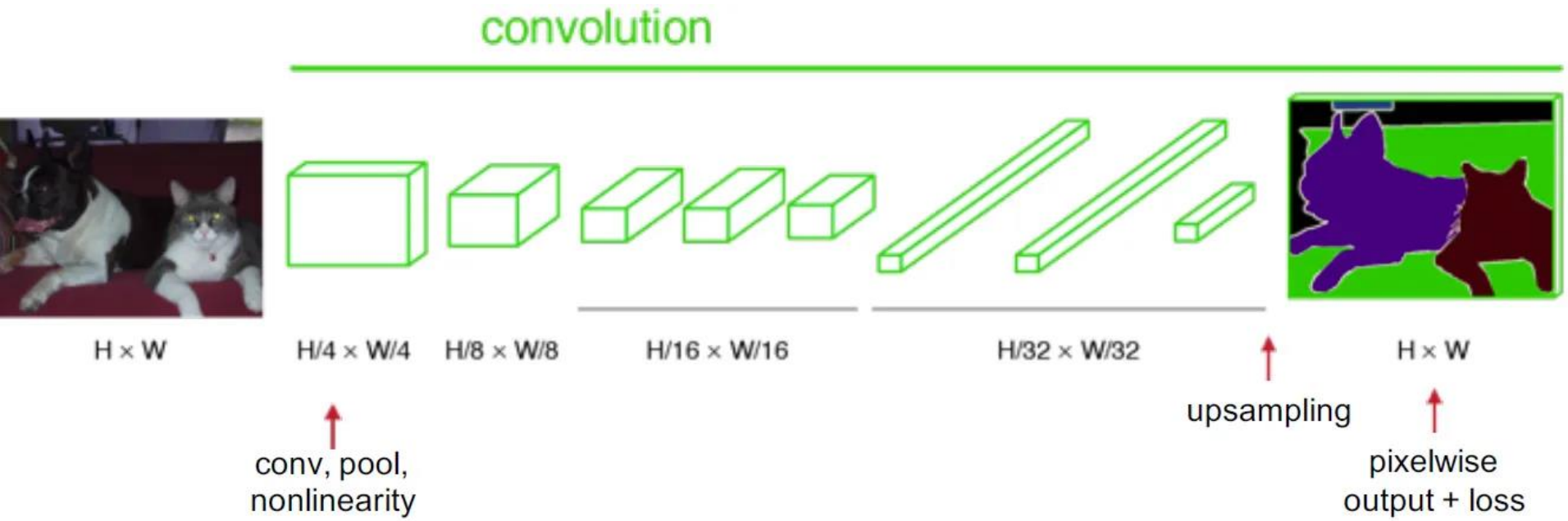
- The output has a size smaller than the input image (due to the max pooling)



STUDENTS-HUB.com
Uploaded By: anonymous

From Image Classification to Semantic Segmentation

- If we upsample the output above, then we can calculate the pixelwise output (label map)



Uploaded By: anonymous

From Image Classification to Semantic Segmentation

- The **number of channels in the final output layer** of a Fully Convolutional Network (FCN) corresponds to the number of **semantic classes** in the segmentation task, including the background class if applicable.
- **How this works:**
  - After upsampling to match the input resolution, the network outputs a H×W×C tensors.
  - Each of the CCC channels is computed as:

$$\text{Channel}_c(x, y) = \sum_{k=1}^{K} \text{Feature}_k(x, y) \cdot W_{k,c} + b_c$$

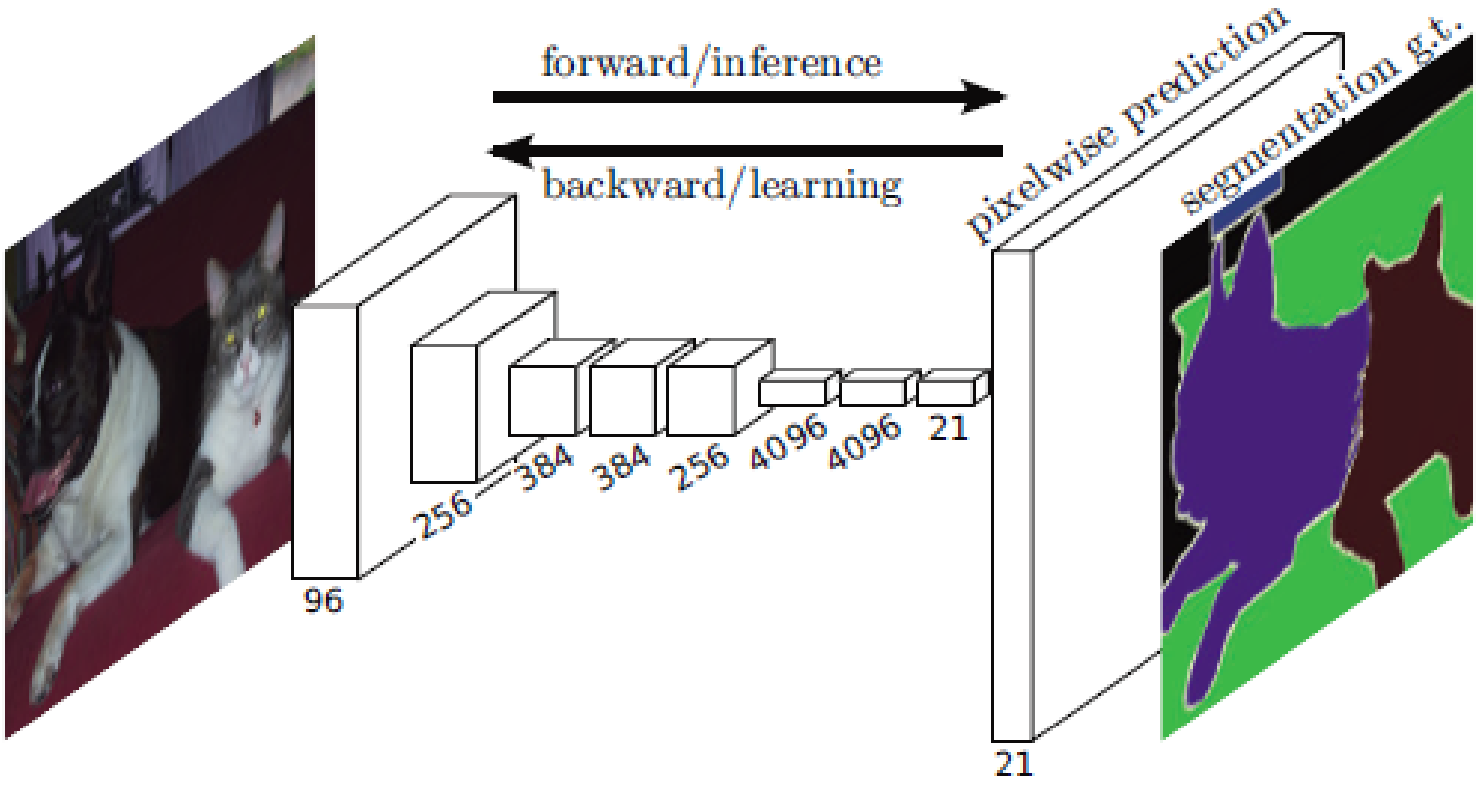
    - K: Number of feature channels from the previous layer.
    - $W_{k,c}$: Weight of the k-th feature map for class c.
    - b_c: Bias for class c.
    - (x, y): Spatial location in the feature map.
  - This operation is a 1x1 convolution that combines features into class scores for each pixel.
  - A softmax activation is applied convert the raw scores into probabilities for each class at each pixel.
  - Do softmax-cross entropy between the pixelwise predictions and segmentaion ground truth

STUDENTS-HUB.com Uploaded By: anonymous

From Image Classification to Semantic Segmentation

- Upsampling at the last step

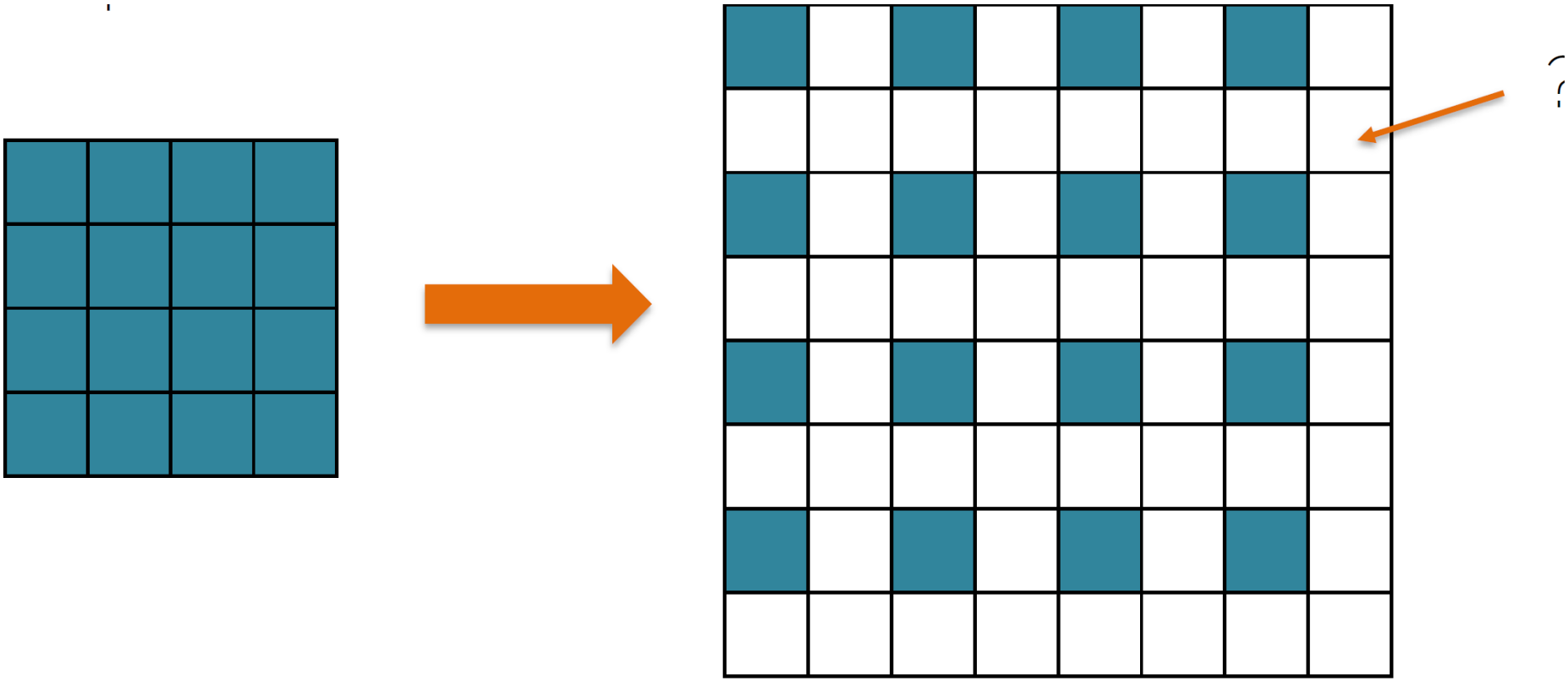
How do we upsample?



Uploaded By: anonymous

From Image Classification to Semantic Segmentation

- Types of upsamplings
  - Interpolation



STUDENTS-HUB.com

Uploaded By: anonymous

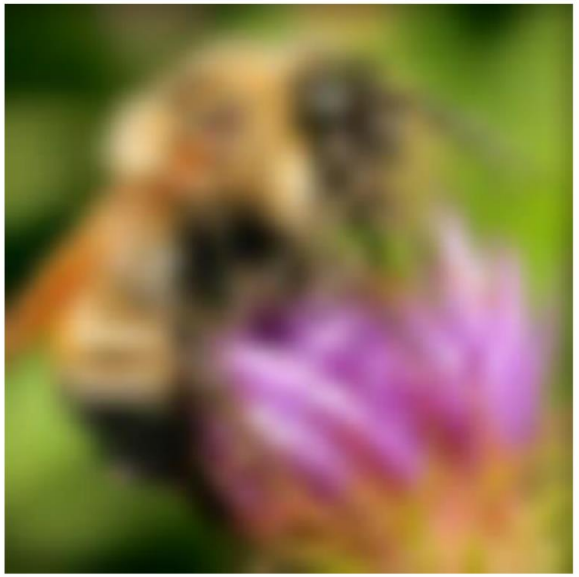
From Image Classification to Semantic Segmentation

- Types of upsamplings
  - Interpolation

Original image  x 10

Nearest neighbor interpolation

Bilinear interpolation



Bicubic interpolation

STUDENTS-HUB.com

Uploaded By: anonymous

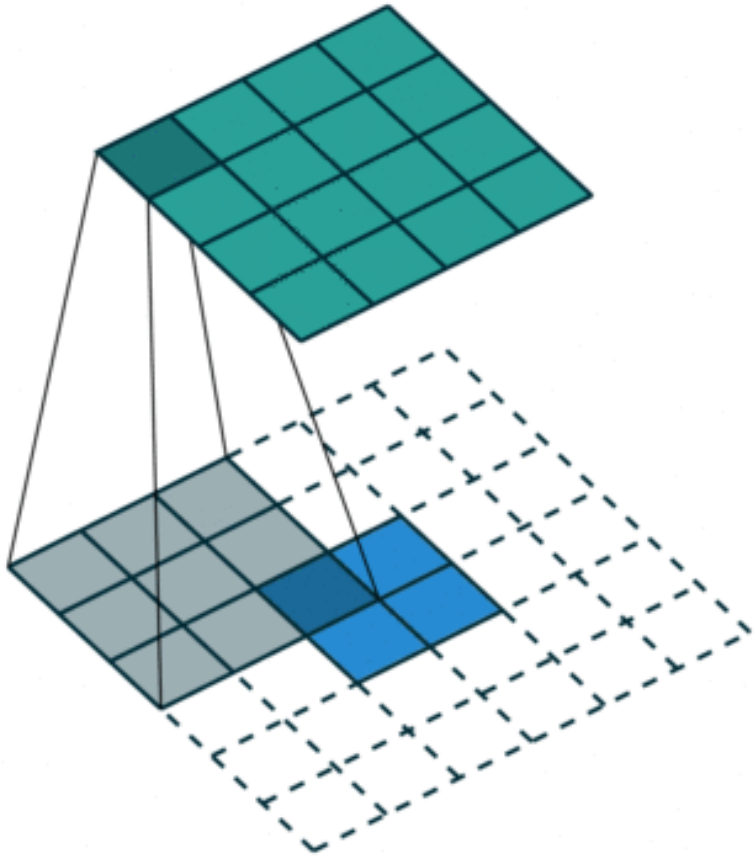
From Image Classification to Semantic Segmentation

- Types of upsamplings
  - Learnable Upsampling: Transposed Convolution
    - **Purpose**: Deconvolution is used to upsample lower-resolution feature maps back to the original input resolution, enabling dense predictions at the pixel level. It helps recover spatial details lost during downsampling in convolutional layers or pooling operations.
    - **Mechanism**: Unlike standard convolution, which reduces spatial dimensions, deconvolution increases the spatial resolution by learning a set of filters to map low-resolution features to a higher resolution.
    - **Trainable Process:** The deconvolution filters are trainable parameters, optimized during backpropagation to reconstruct meaningful spatial information while preserving semantic understanding.
    - **Preserving Structure**: By aligning features with the input image dimensions, deconvolution ensures that the upsampled outputs maintain consistency with the spatial layout of the original image.

STUDENTS-HUB.com Uploaded By: anonymous

From Image Classification to Semantic Segmentation

- Types of upsamplings
  - Learnable Upsampling: Transposed Convolution

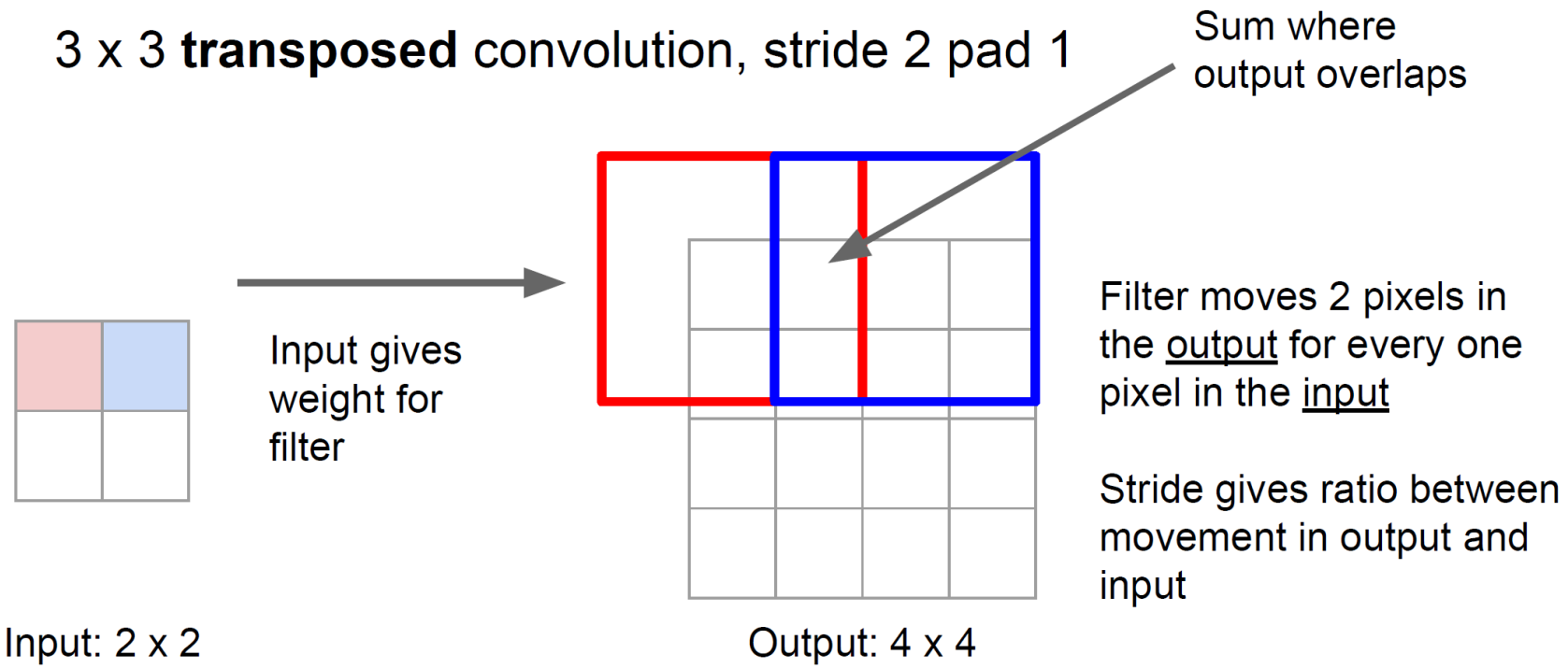


(Blue: Input, Green: Output)

STUDENTS-HUB.com Uploaded By: anonymous

From Image Classification to Semantic Segmentation

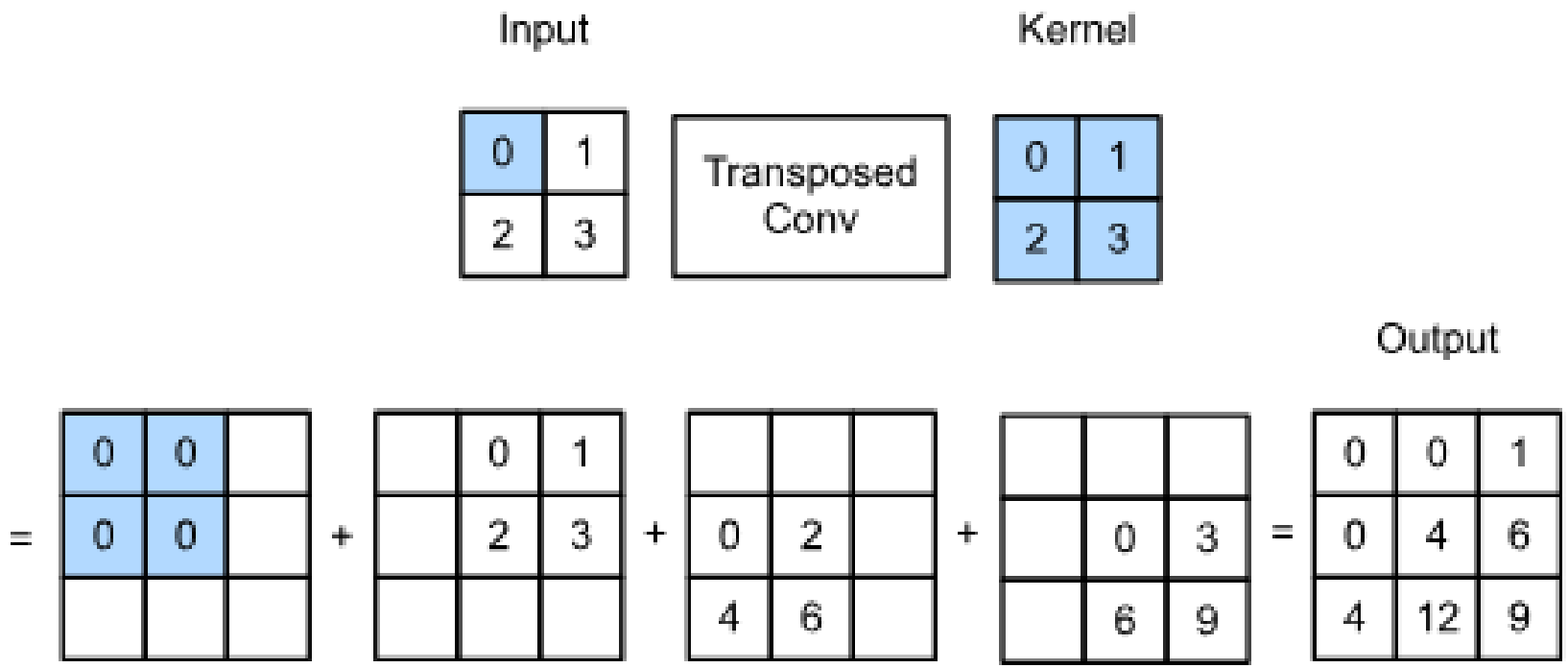
- Types of upsamplings
  - Learnable Upsampling: Transposed Convolution



STUDENTS-HUB.com
Uploaded By: anonymous

From Image Classification to Semantic Segmentation

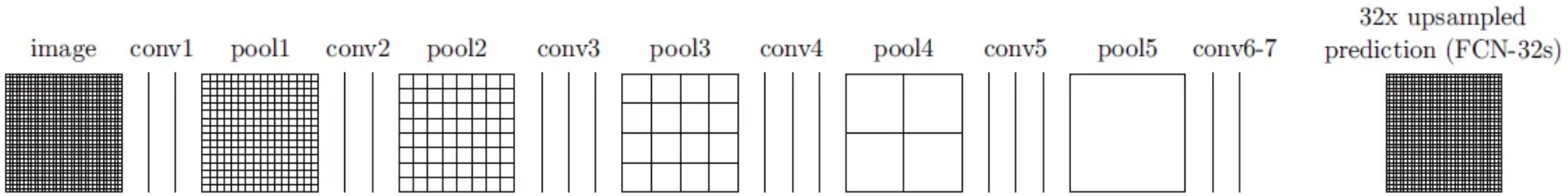
- Types of upsamplings
  - Learnable Upsampling: Transposed Convolution



STUDENTS-HUB.com
Uploaded By: anonymous

From Image Classification to Semantic Segmentation

- After going through conv7 as below, the output size is small, then 32× upsampling is done to make the output have the same size of input image. But it also makes the output label map rough. And it is called **FCN-32s**

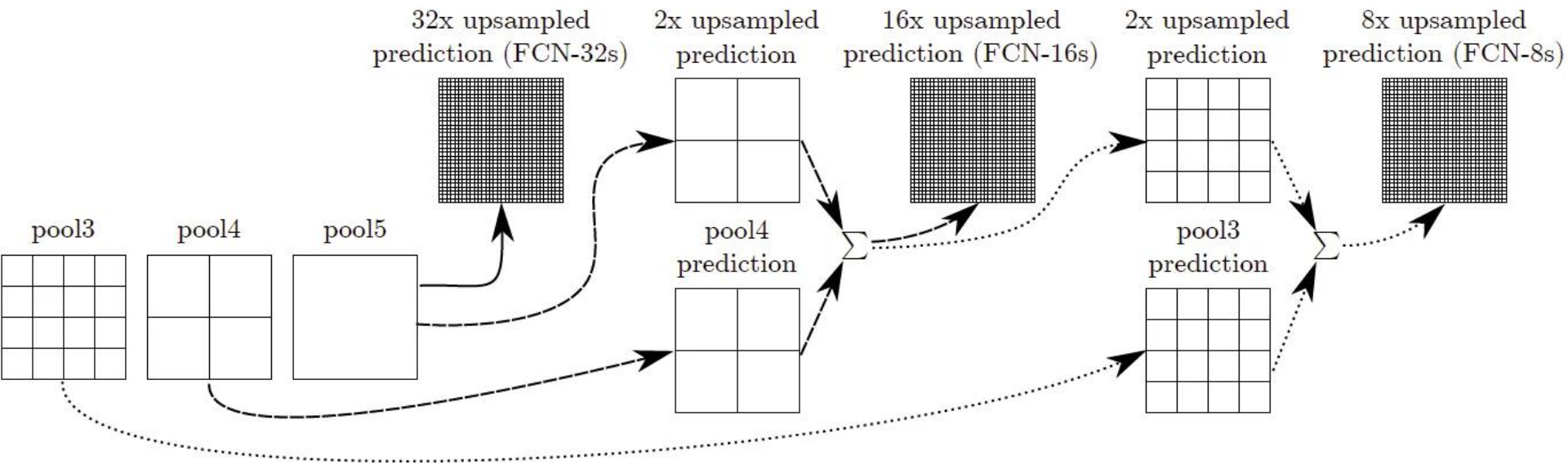


- **Deep features can be obtained when going deeper, spatial location information is also lost when going deeper.** That means output from shallower layers have more location information. If we combine both, we can enhance the result.
- To combine, we **fuse the output (by element-wise addition)**

STUDENTS-HUB.com

Uploaded By: anonymous

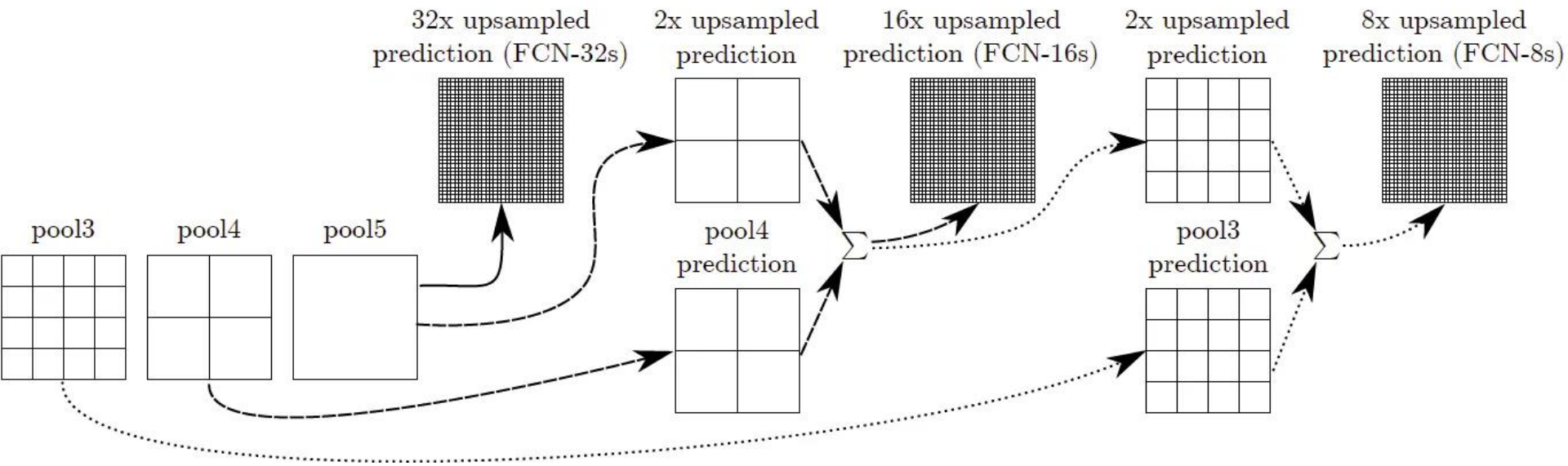
From Image Classification to Semantic Segmentation



- **FCN-16x**
  - The output from pool5 is 2× upsampled and fuse with pool4 and perform 16× upsampling
  - Fusion is performed by **element-wise addition** of the upsampled final layer and the intermediate layer

Uploaded By: anonymous

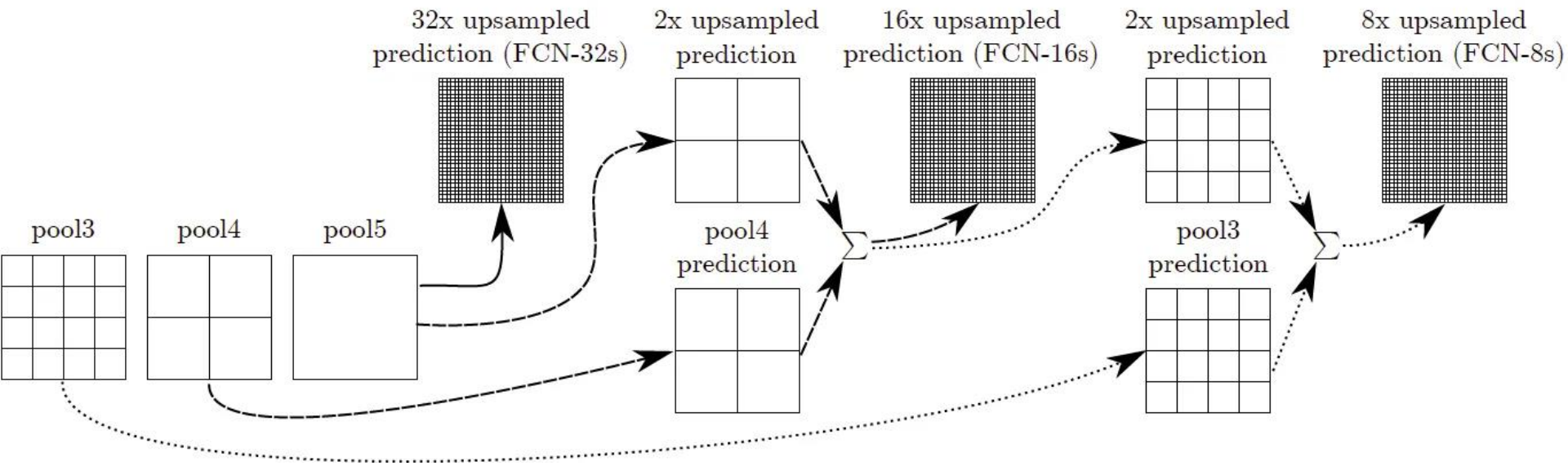
From Image Classification to Semantic Segmentation



- **FCN-8x**:
  - Combines the **final feature map** with two intermediate feature maps pool4 and pool3
  - The output from the deepest layer is first upsampled by 2x and fused with pool4.
  - The resulting feature map is then upsampled by 2x again and fused with pool3, which has even higher spatial resolution (8x smaller than the input image).
  - Finally, the combined map is upsampled 8x to match the input resolution.

STUDENTS-HUB.com Uploaded By: anonymous

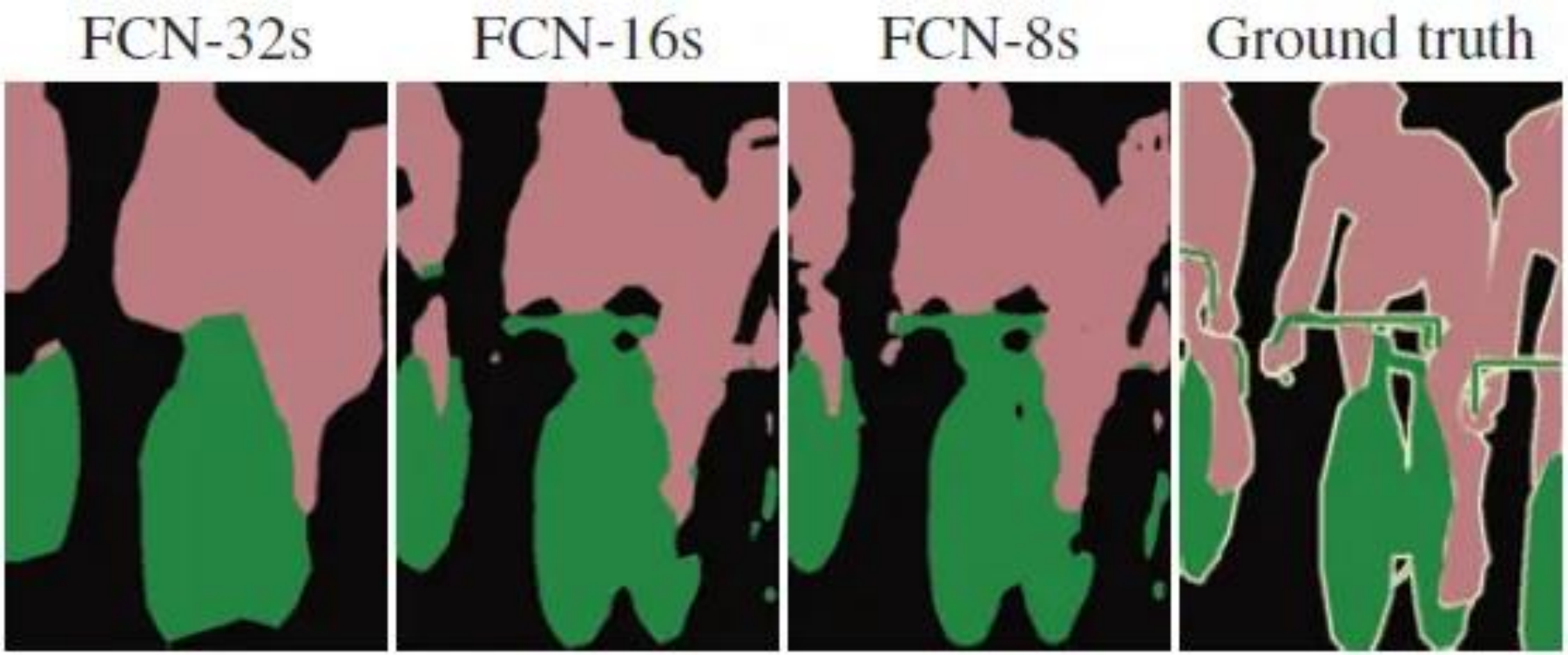
From Image Classification to Semantic Segmentation



- Hierarchical training where the network is initially trained only based on high level features and then finetuned based on middle and low-level features.
- **Fine-Tuning from Pretrained Models**
  - FCNs are often initialized using pretrained models (e.g., VGG-16 or ResNet) trained on large-scale image classification datasets like **ImageNet**.
  - Pretrained weights for the encoder (convolutional layers) are used as a starting point, and the decoder layers (deconvolution layers) are initialized randomly.
  - Fine-tuning is performed by updating all layers (or only the decoder layers) on the semantic segmentation dataset.

STUDENTS-HUB.com Uploaded By: anonymous

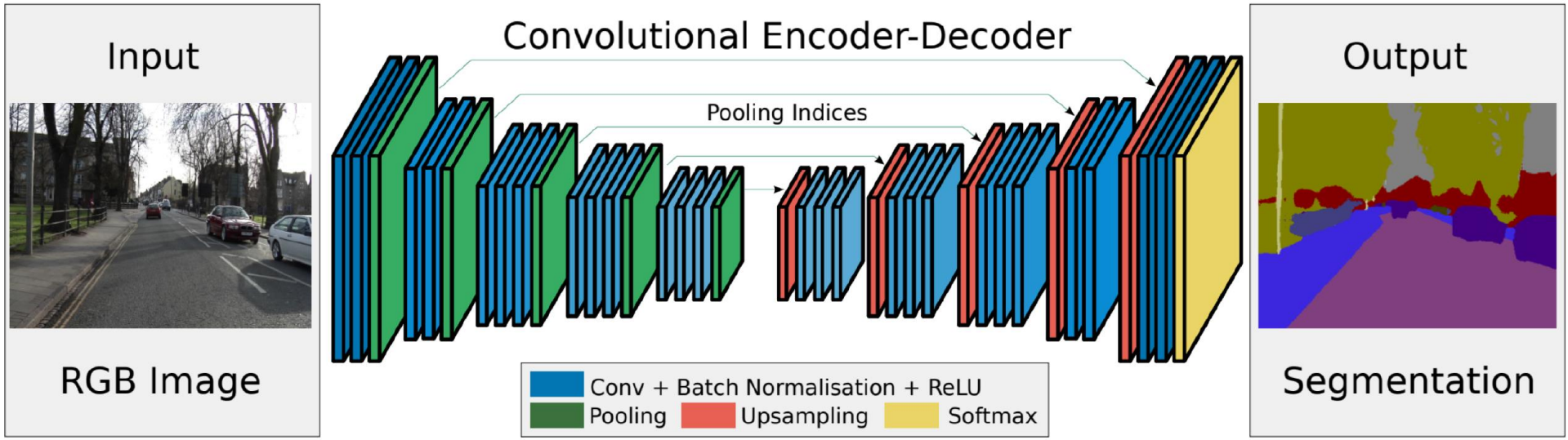
From Image Classification to Semantic Segmentation



STUDENTS-HUB.com
Uploaded By: anonymous

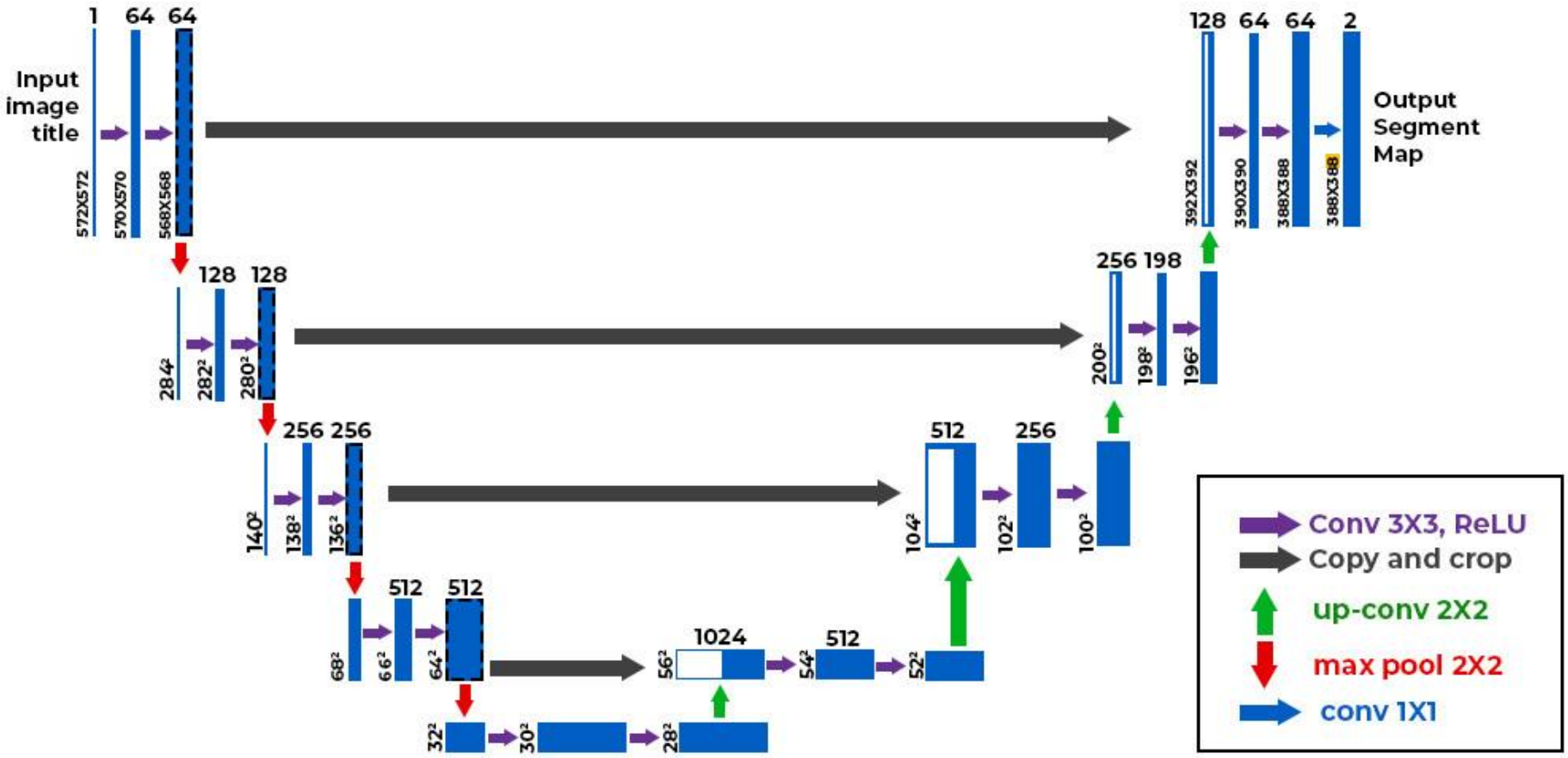
Autoencoder-style architecture - SegNet

- Step-wise upsampling
  - **Encoder**: normal convolutional filters + pooling
  - **Decoder**: Upsampling + convolutional filters
    - The convolutional filters in the decoder are learned using backprop and their goal is to refine the upsampling



Uploaded By: anonymous

Skip Connection: U-Net

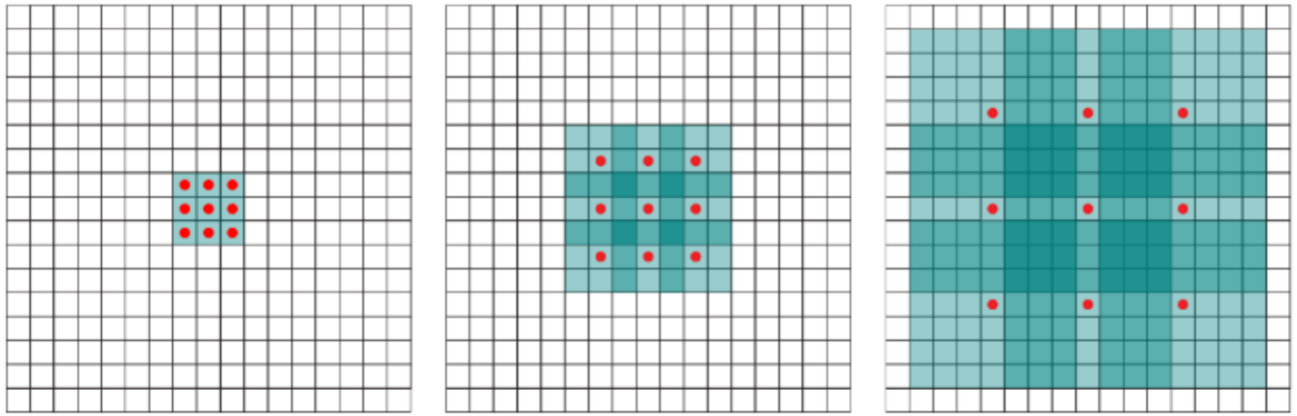


Uploaded By: anonymous

Semantic Segmentation: Challenges

Reduced feature resolution

- As images pass through a deep neural network, especially in the early layers, the spatial resolution of the feature maps typically decreases due to operations like **pooling** and **striding**.
- This reduction in resolution can lead to the loss of fine-grained spatial information, which is crucial for pixel-level predictions in segmentation tasks.
- Solution: Dilated/Atrous Convolutions
  - Type of convolutional operation that introduces **gaps** or **dilations** between the kernel elements.
  - This technique allows the convolution to cover a larger receptive field without increasing the number of parameters or the computational cost significantly.

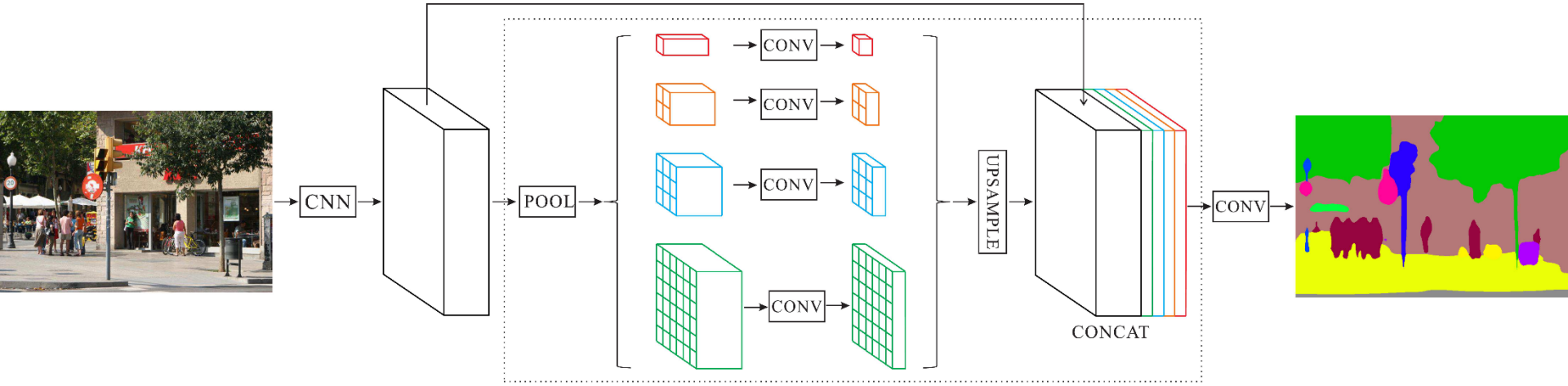


STUDENTS-HUB.com

Uploaded By: anonymous

Semantic Segmentation: Challenges

- **Objects exist at multiple scales**
  - Solution: Pyramid pooling, as in detection.
    - **Captures multi-scale context** by applying pooling at different spatial resolutions.
    - **Enhances feature representation** by combining pooled features from multiple scales.
    - **Improves segmentation accuracy** by preserving fine details and capturing larger context.

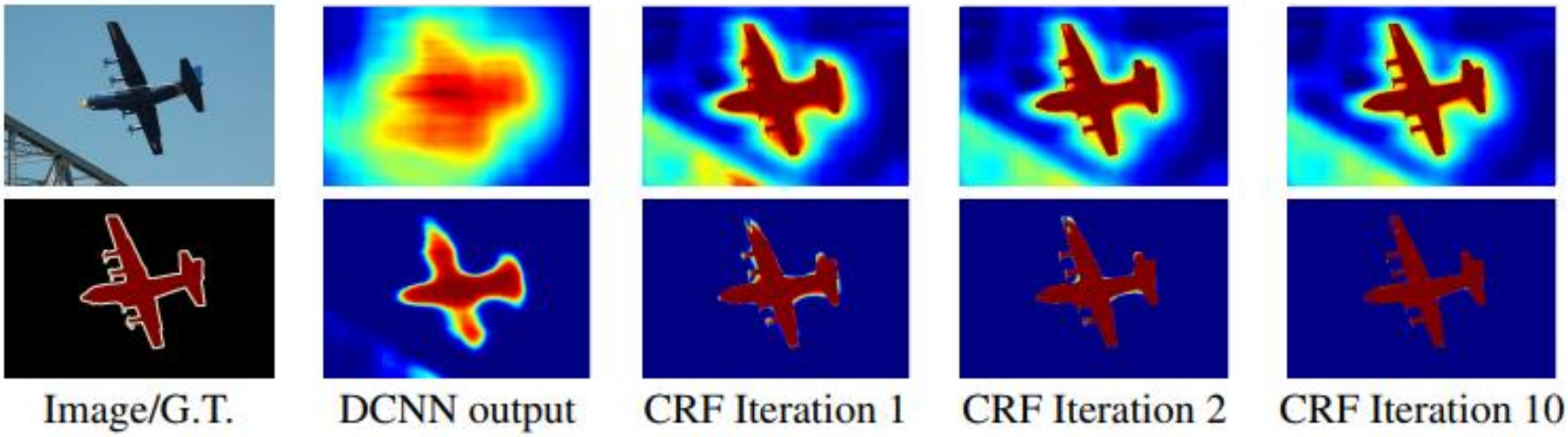


(a) Input Image (b) Feature Map (c) Pyramid Pooling Module (d) Final Prediction

STUDENTS-HUB.com Uploaded By: anonymous

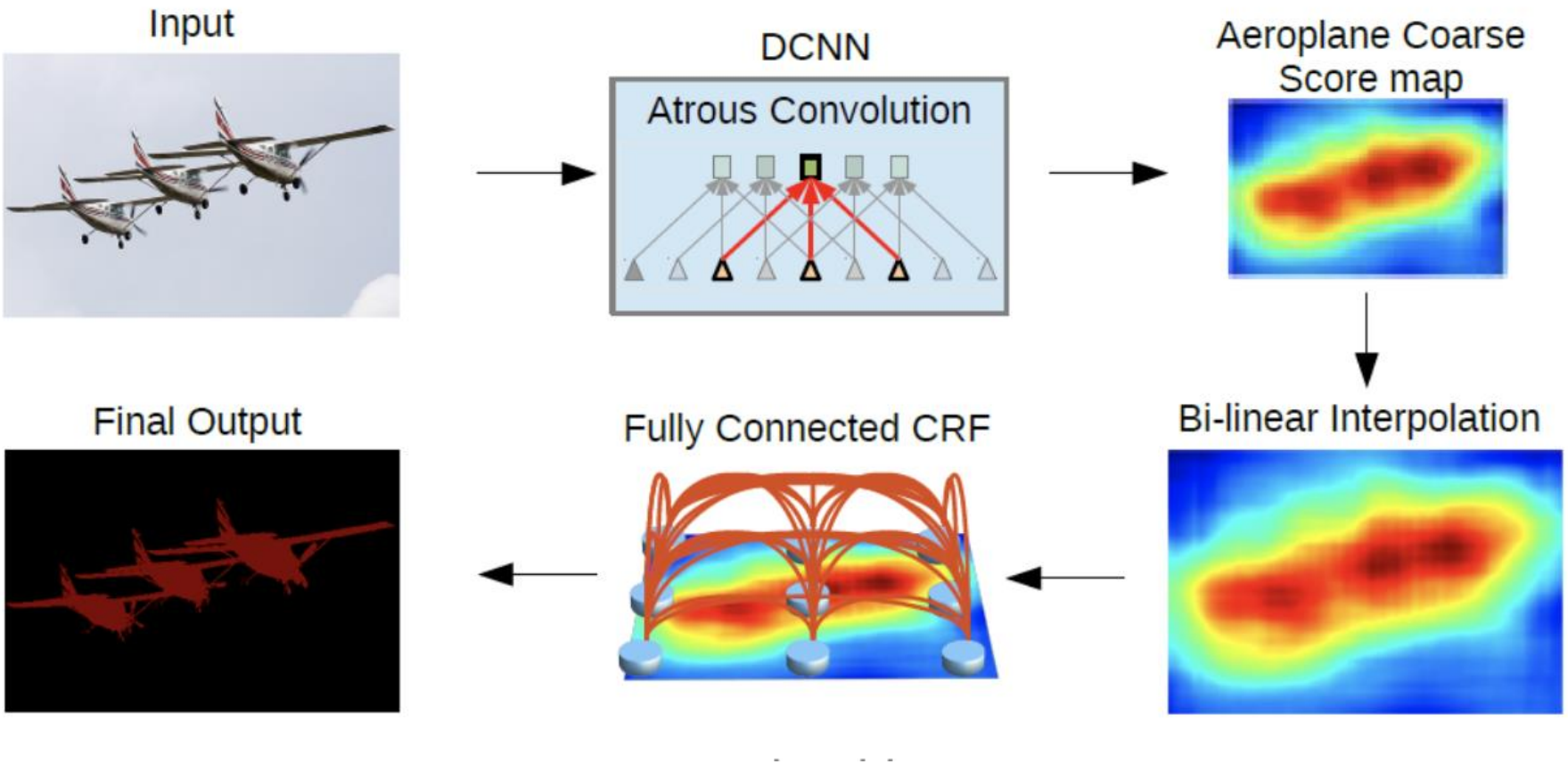
Semantic Segmentation: Challenges

- Poor localization of the edges
  - Solution: Refinement with Conditional Random Field (CRF)
    - Conditional Random Fields (CRFs) work by modeling the **conditional dependencies** between pixels in an image, which helps refine the output of segmentation models like Fully Convolutional Networks (FCNs).
    - CRFs refine the output of segmentation models like FCNs by modeling spatial dependencies between neighboring pixels.
    - They use unary potentials (from the FCN output) and pairwise potentials (based on pixel similarity) to optimize the segmentation and improve edge localization.



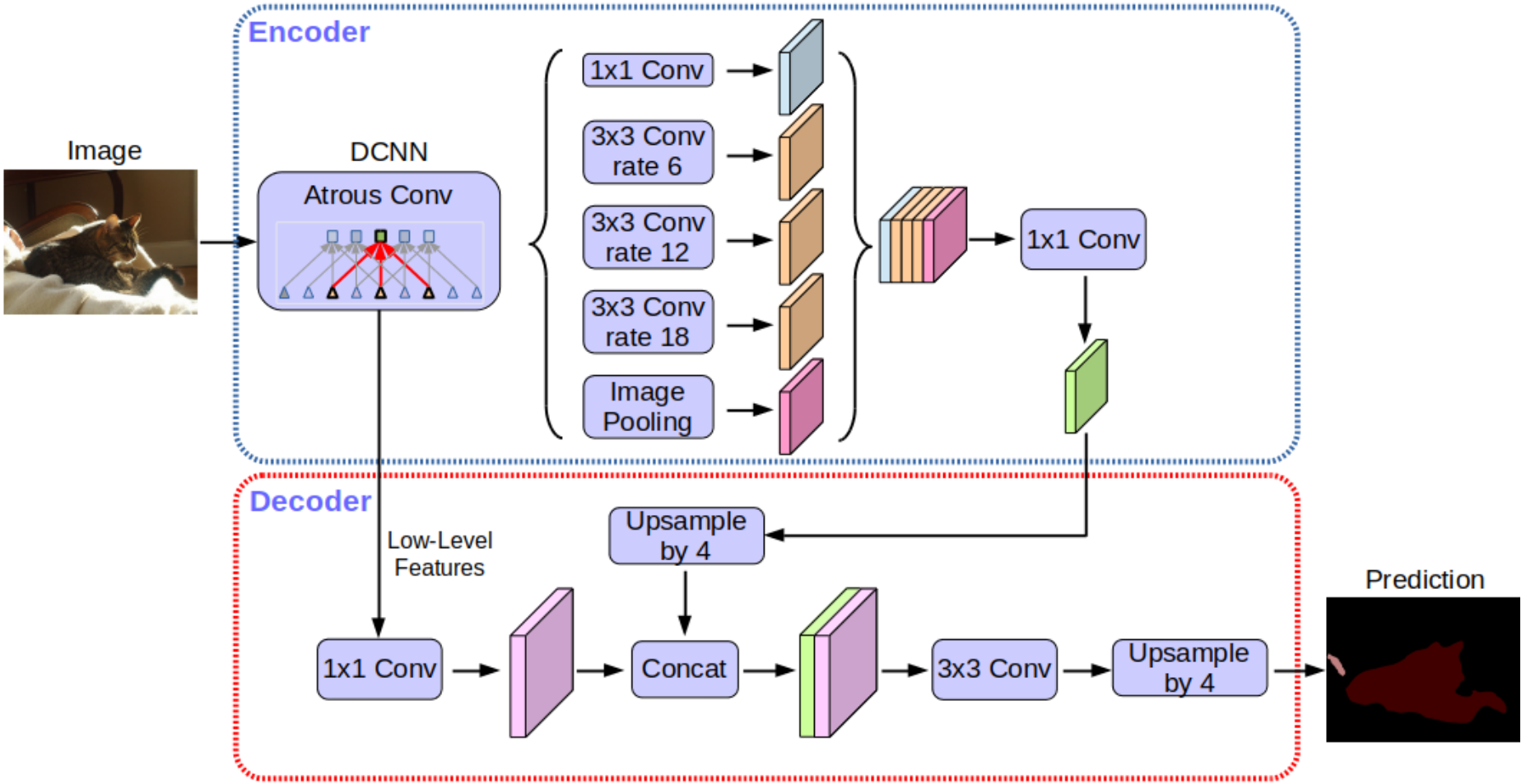
STUDENTS-HUB.com
Uploaded By: anonymous

Other Networks: **DeepLab**



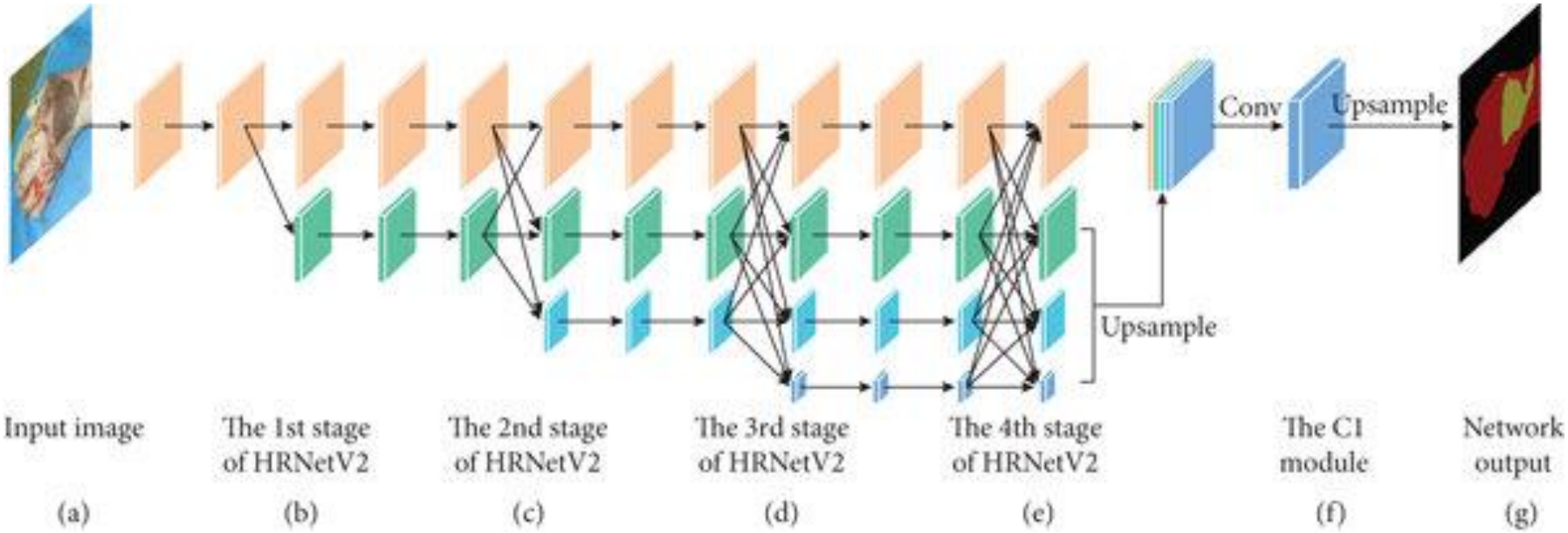
Uploaded By: anonymous

Other Networks: **DeepLabV3**



Uploaded By: anonymous

Other Networks: HRNetV2



Uploaded By: anonymous

Datasets

Pascal VOC 2012:

9993 natural images divided into 20 classes.

Cityscapes:

25K urban-street images divided into 30 classes.

ADE20K:

25K (20 stands for 20K training) scene-parsing images divided into 150 classes.

Mapillary Vistas:

25K street level images, divided into 152 classes.

- Models are often pre-trained in the large MS-COCO dataset, before finetuned to the specific dataset.

STUDENTS-HUB.com

Uploaded By: anonymous

Metrics: mean intersection over union (mIoU)

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

MIoU simply computes the IoU for each class and then computes the mean of those values.

Another widely used metric is the pixel accuracy (ratio of pixels classified correctly).

STUDENTS-HUB.com

Uploaded By: anonymous

Some Comparisons

Model	Inference Time (ms)	MB Allocated
UNet	195	4389
LinkNet	171	4389
PSPNet	89	3877
DeepLabv2	344	6689
DeepLabv2(msc)	623	5413
DeepLabv3+	117	4901

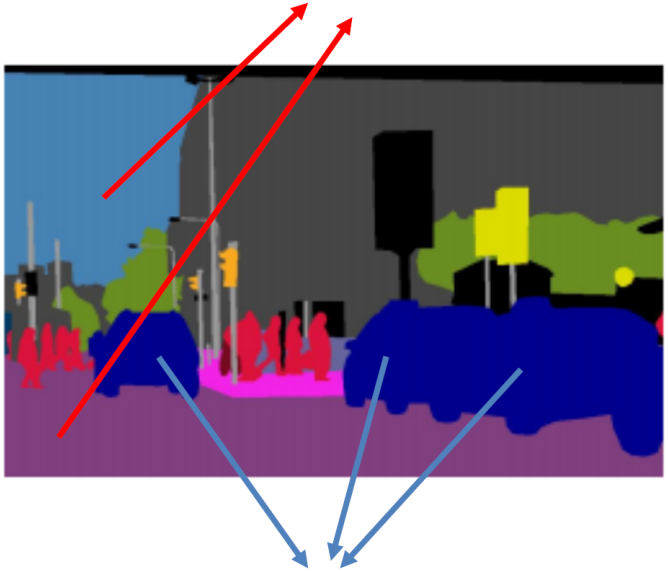
Performance metrics	UNet		SegNet		DeeplabV3	
	Without backbone	With backbone	Without backbone	With backbone	Without backbone	With backbone
Accuracy	0.574	0.653	0.564	0.673	0.681	0.763
Precision	0.590	0.657	0.563	0.678	0.6855	0.761
Recall	0.574	0.653	0.566	0.673	0.681	0.763
F1 score	0.573	0.646	0.559	0.672	0.674	0.756
Jaccard Coefficient (IOU)	0.411	0.500	0.399	0.528	0.5308	0.626
Mean IOU	0.306	0.323	0.290	0.407	0.410	0.520

STUDENTS-HUB.com

Uploaded By: anonymous

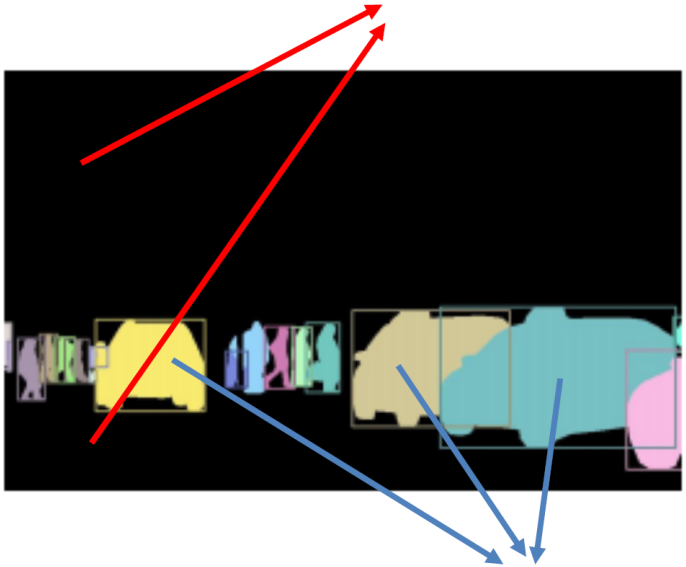
Instance segmentation

Label every pixel, including the background (sky, grass, road)



Do not differentiate between the pixels coming from instances of the same class

Do not label pixels coming from uncountable objects (sky, grass, road)



Differentiate between the pixels coming from instances of the same class

STUDENTS-HUB.com

Uploaded By: anonymous

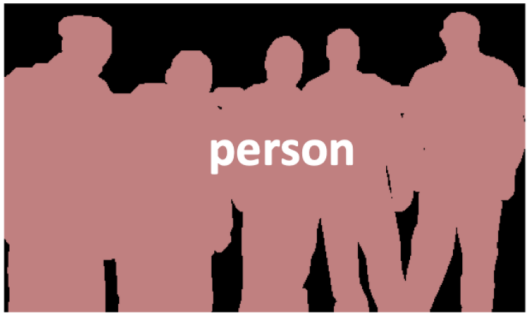
Instance segmentation methods

Proposal-based

FCN-based

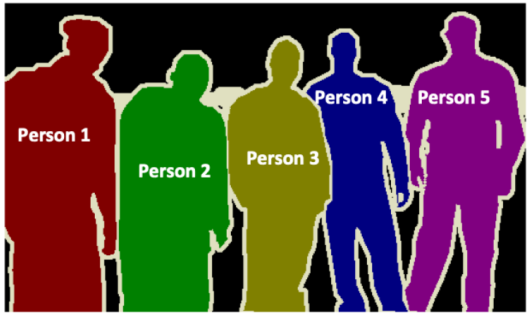
1. Proposals

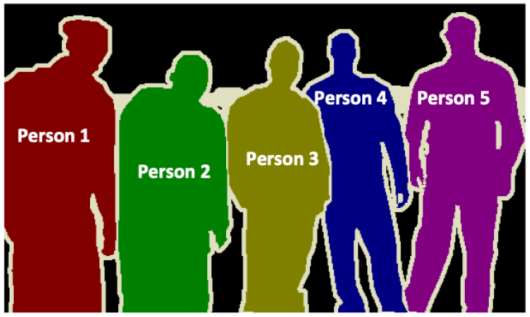
vs.



1. Semantic segmentation

2. Assign a class





2. Find instances

STUDENTS-HUB.com

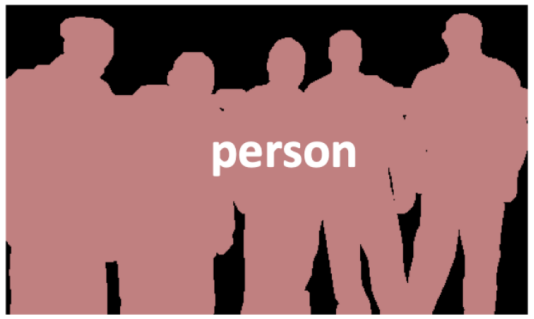
Uploaded By: anonymous

IS: the best of both worlds

Proposal-based

FCN-based

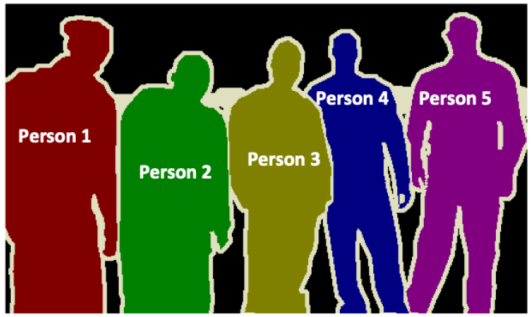
1. Proposals

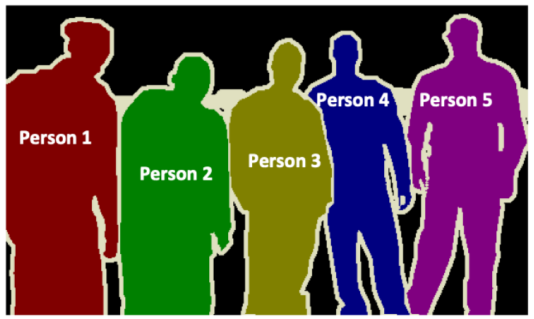


1. Semantic segmentation

+

2. Assign a class





2. Find instances

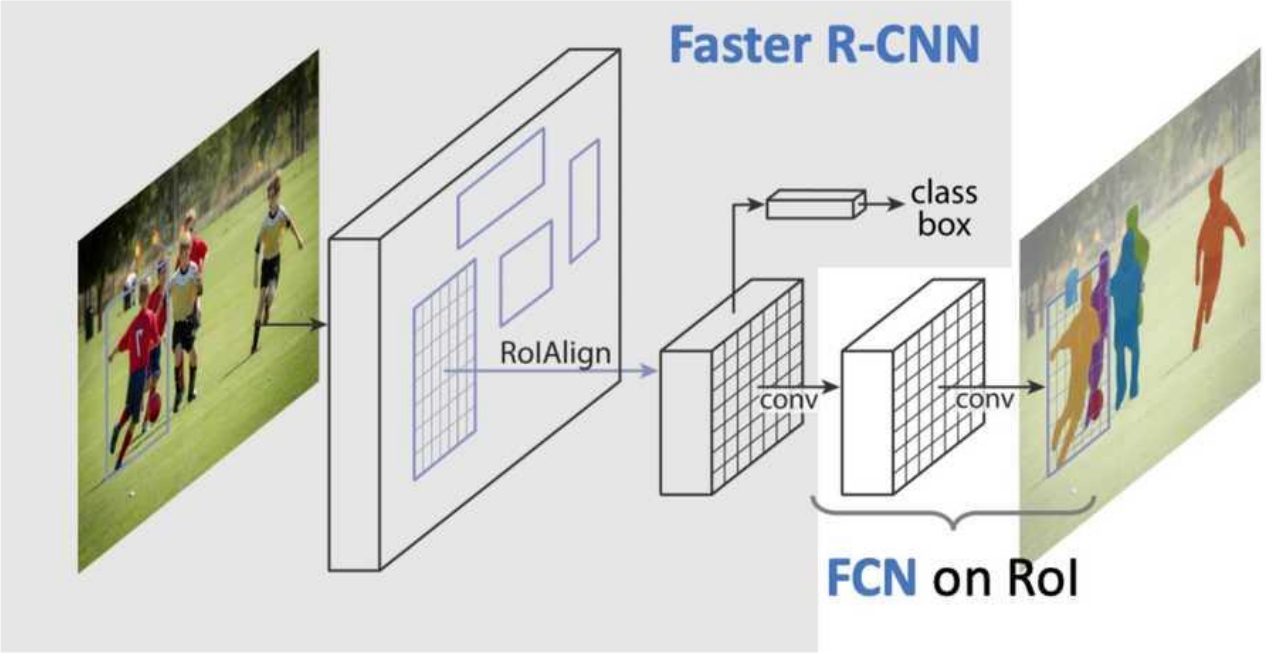
STUDENTS-HUB.com

Uploaded By: anonymous

Mask-RCNN

- Mask R-CNN is an extension of the Faster R-CNN architecture designed for **instance segmentation**, which not only detects objects but also generates pixel-level masks for each object.

Mask R-CNN = Faster R-CNN + FCN on ROIs



Uploaded By: anonymous

Mask R-CNN: qualitative results

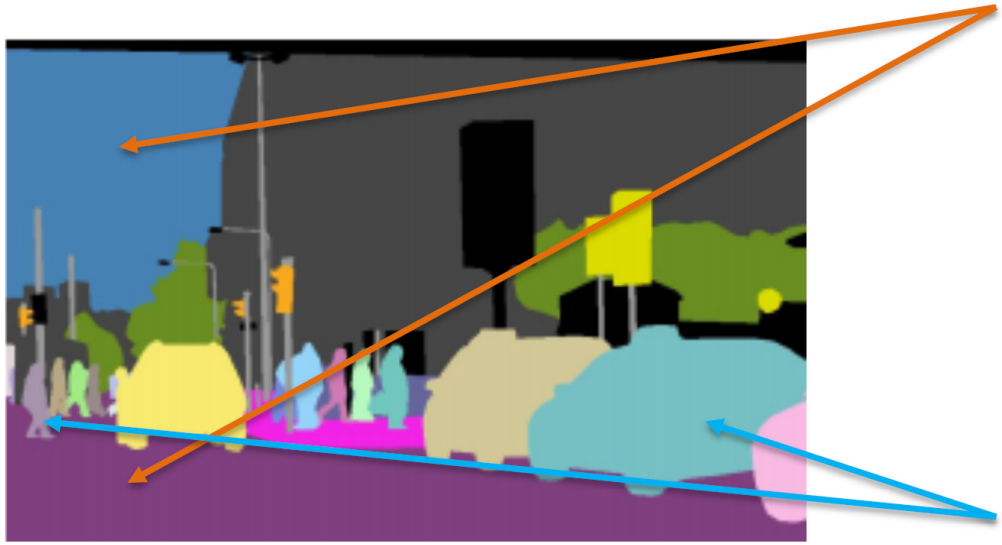
Some Comparisons

Network	Backbone	Training time (h)	mPrecision (%)	mRecall (%)	mF1 (%)	mIoU (%)
U-Net	VGG16	**0.06**	74.34	65.53	69.66	54.71
U-Net	ResNet-50	0.16	83.15	75.40	79.09	64.82
PSPNet	MobileNetV2	0.19	74.12	68.22	71.05	56.77
PSPNet	ResNet-50	0.10	82.82	77.75	80.20	66.78
DeepLabv3+	MobileNetV2	0.29	80.01	73.41	76.57	61.21
DeepLabv3+	Xception	0.74	78.85	73.48	76.07	60.14
Mask R-CNN	ResNet-50	0.43	46.97	37.00	41.39	27.56
Mask R-CNN	ResNet-101	0.48	47.83	40.39	43.80	28.04
HRNetV2	HRNetV2-W18	0.16	82.91	74.86	78.68	64.52
HRNetV2	HRNetV2-W32	0.16	**84.00**	**81.86**	**82.92**	**70.94**
HRNetV2	HRNetV2-W48	0.17	83.43	78.67	80.93	68.37

STUDENTS-HUB.com

Uploaded By: anonymous

Panoptic segmentation



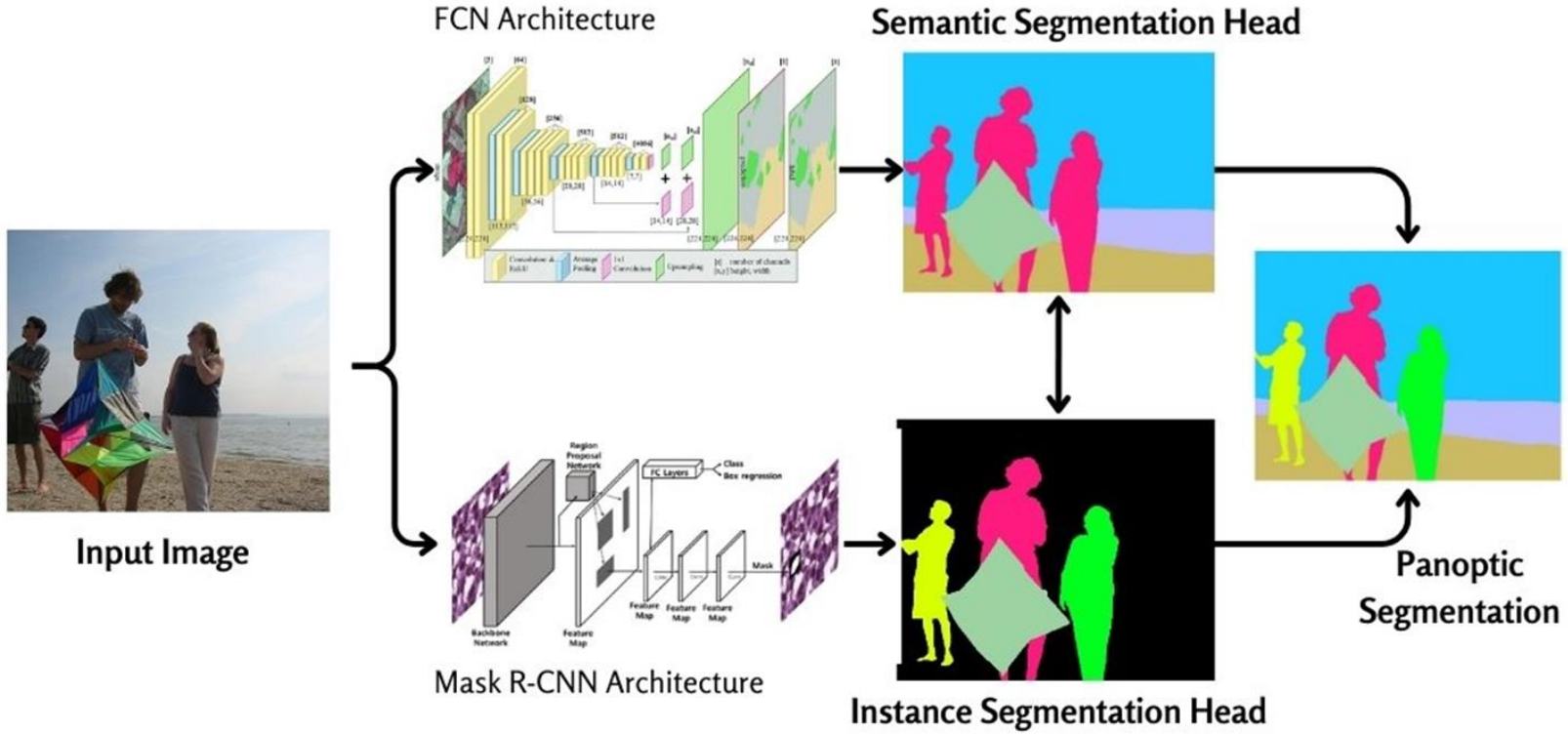
It gives labels to uncountable objects called "stuff" (sky, road, etc), similar to FCN-like networks.

It differentiates between pixels coming from different instances of the same class (countable objects) called "things" (cars, pedestrians, etc).

STUDENTS-HUB.com

Uploaded By: anonymous

Panoptic segmentation

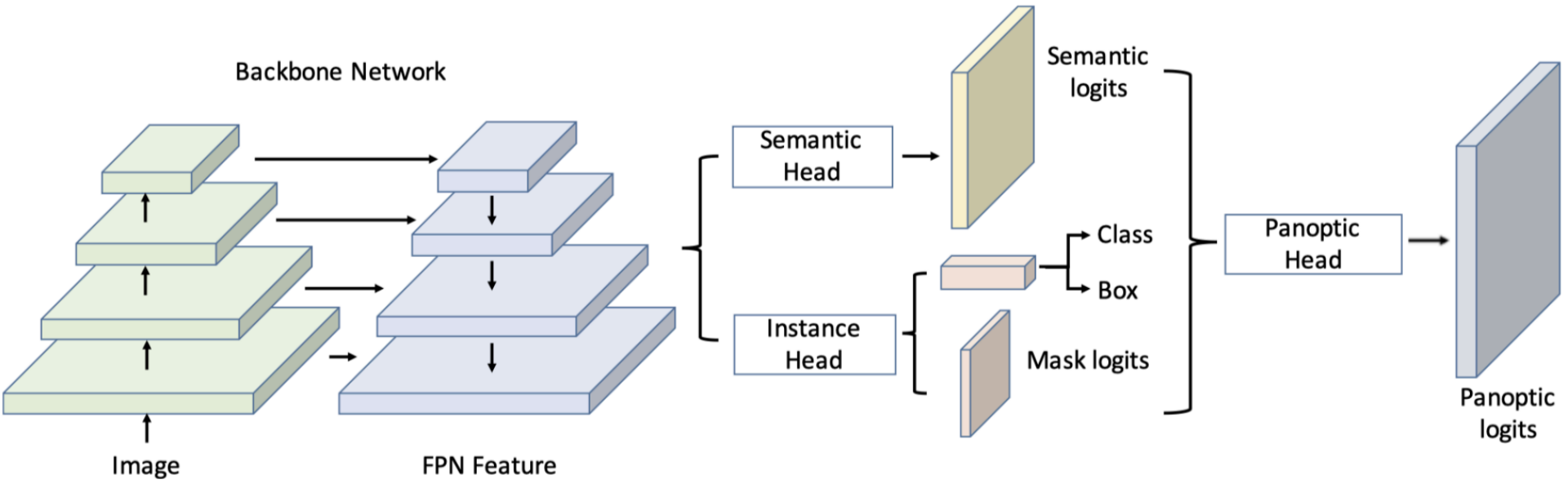


- **Problem:** some pixels might get classified as stuff from FCN network, while at the same time being classified as instances of some class from Mask R-CNN (conflicting results)!
- **Solution:** Parametric-free panoptic head which combines the information from the FCN and Mask R-CNN, giving final predictions.

STUDENTS-HUB.com
Uploaded By: anonymous

UPSNet

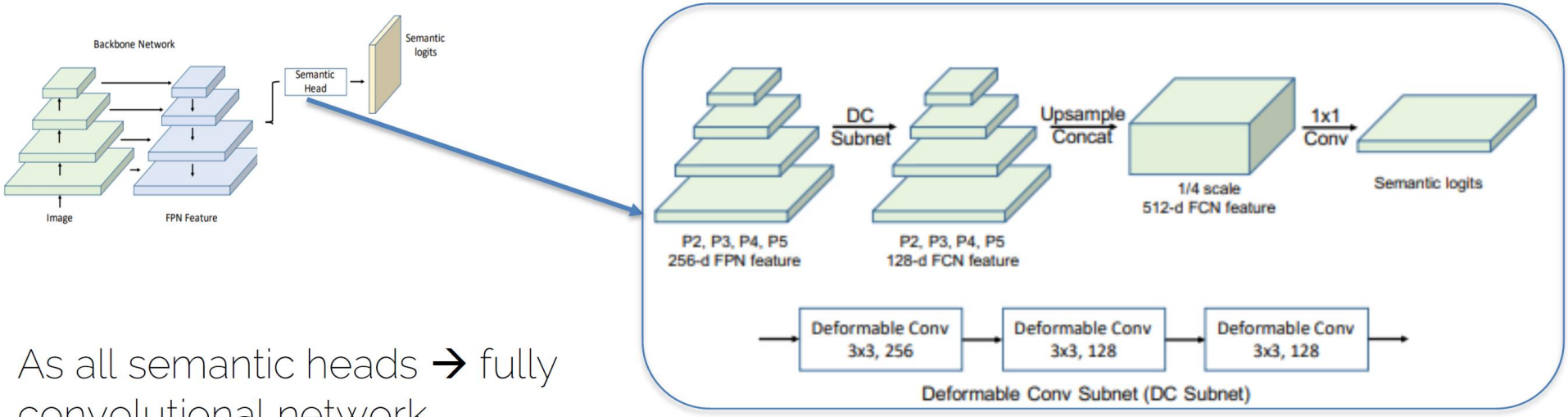
- **Unified Architecture:** By sharing a backbone and combining outputs, UPSNet achieves a more coherent and efficient solution compared to separate networks for semantic and instance segmentation.
- **End-to-End Training:** UPSNet can be trained end-to-end, allowing for joint optimization of the semantic and instance segmentation branches.



STUDENTS-HUB.com
Uploaded By: anonymous

UPSNet

- **Semantic Segmentation Head:** This branch focuses on classifying each pixel into its semantic class. It often employs techniques like deformable convolutions to capture intricate object shapes and boundaries.



As all semantic heads → fully convolutional network.

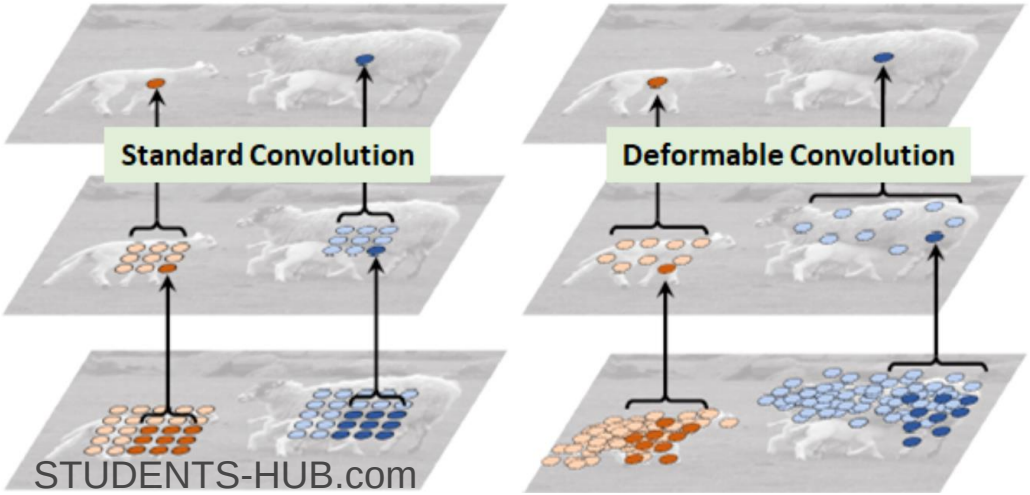
New: deformable convolutions!

STUDENTS-HUB.com

Uploaded By: anonymous

UPSNet

- In standard convolutional neural networks (CNNs), the sampling grid for convolution is fixed and regular.
- Deformable convolutions address this limitation by introducing **learnable offsets** to the regular sampling grid.
  - **Offset Prediction:** A separate branch in the network predicts offset vectors for each sampling location. These offsets indicate how much each sampling location should be shifted from its original position.
- Deformable convolutions: generalization of dilated convolutions when you learn the offset

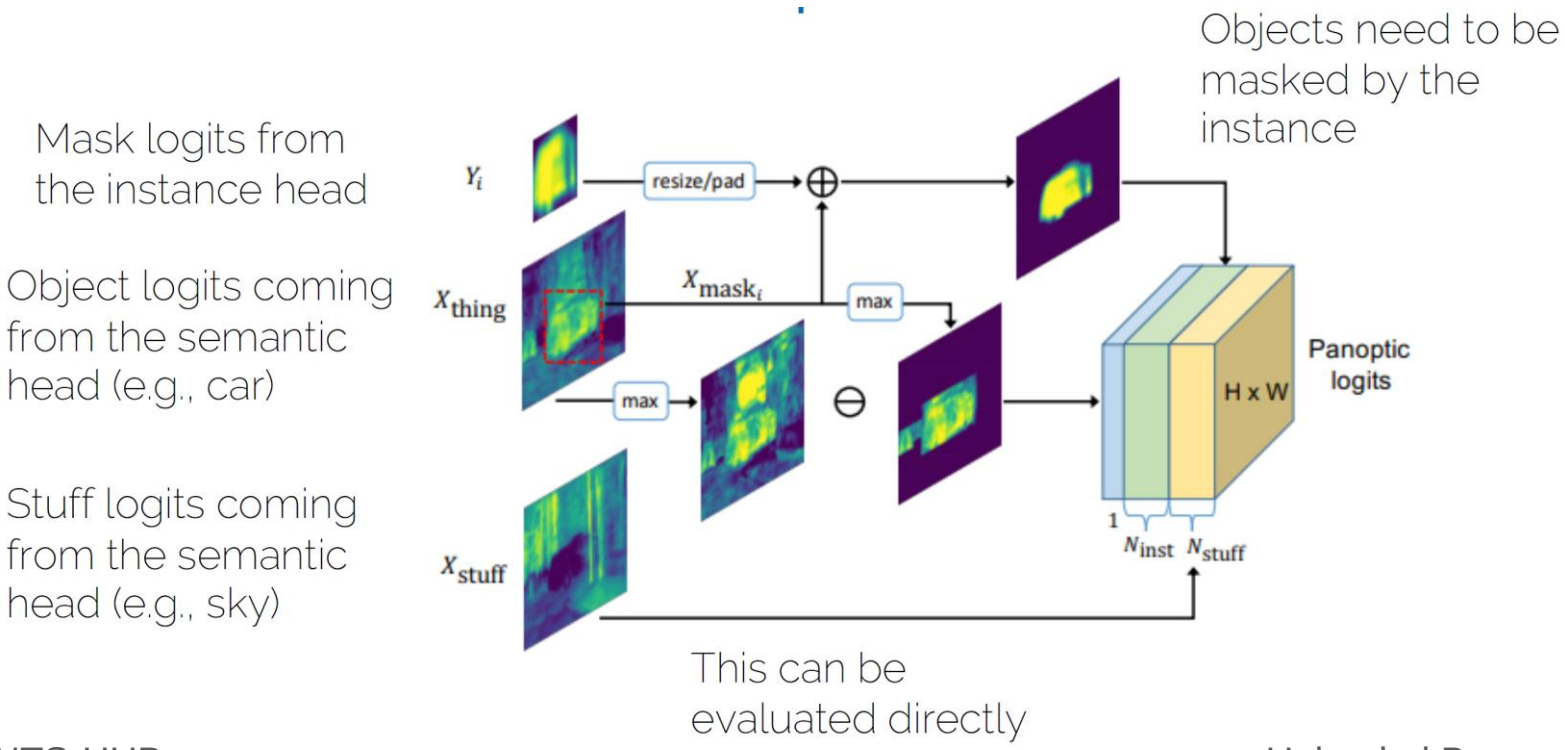


The deformable convolution will pick the values at different locations for convolutions conditioned on the input image of the feature maps.

STUDENTS-HUB.com
Uploaded By: anonymous

UPSNet

- **Panoptic Head:** This is a unique component of UPSNet.
  - It takes the outputs from the semantic and instance segmentation heads and combines them to produce a final panoptic segmentation map.
  - This head resolves potential conflicts between the two subtasks and assigns a unique ID to each instance.



Uploaded By: anonymous

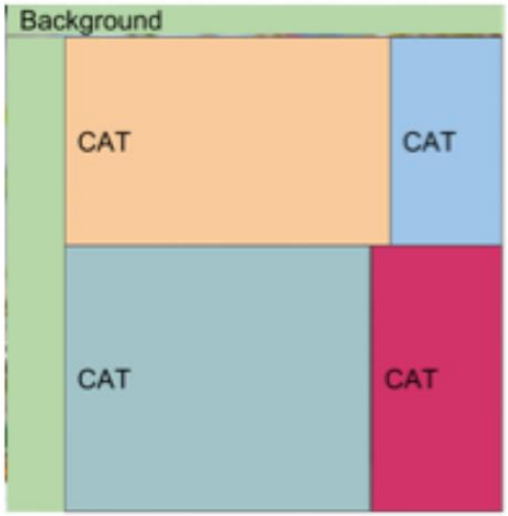
UPSNet

- **Panoptic Logits:** These are the raw output values from the network for each pixel, representing the predicted probabilities for different classes and instances.
- **Softmax:** Softmax is applied to these logits. This converts the raw values into probabilities that sum to 1 across all classes and instances.
- **Class Determination:**
  - If the maximum probability after softmax occurs within the "stuff" class channels (e.g., sky, road, grass), then the pixel is classified as belonging to one of these stuff classes.
  - If the maximum probability is found in the instance channels, then the index of that channel corresponds to the instance ID of the pixel.

STUDENTS-HUB.com Uploaded By: anonymous

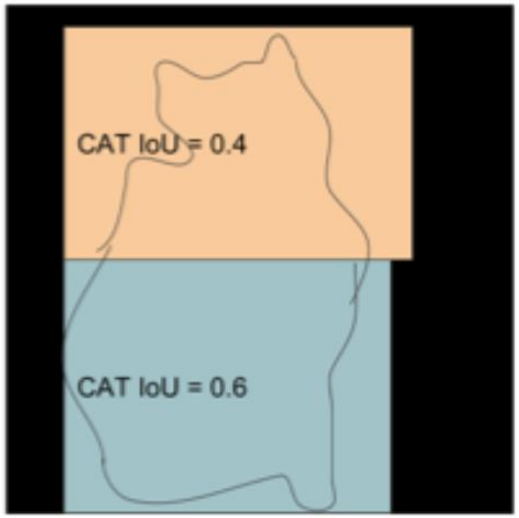
Metrics: Panoptic quality

- As in detection, we have to “match ground truth and predictions. In this case we have segment matching.
- Segment is matched if IoU>0.5. No pixel can belong to two predicted segments.



Predictions

Ground truth



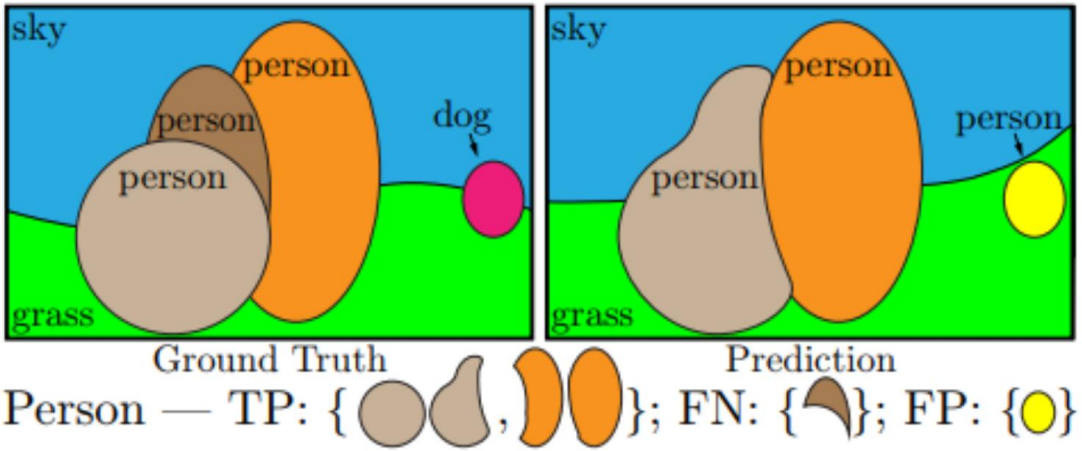
IoU measures

Uploaded By: anonymous

Metrics: Panoptic quality

$$PQ = \underbrace{\frac{\sum_{(p,g)\in \mathrm{TP}} \mathrm{IoU}(p,g)}{|\mathrm{TP}|}}_{\mathrm{SQ}} \underbrace{\frac{|\mathrm{TP}|}{|\mathrm{TP}| + \frac{1}{2}|\mathrm{FP}| + \frac{1}{2}|\mathrm{FN}|}}_{\mathrm{RQ}}$$

TP = True positive, FN = False negative, FP = false positive



SQ: Segmentation Quality = how close the predicted segments are to the ground truth segment

Measures the **average Intersection over Union (IoU)** of correctly matched segments.

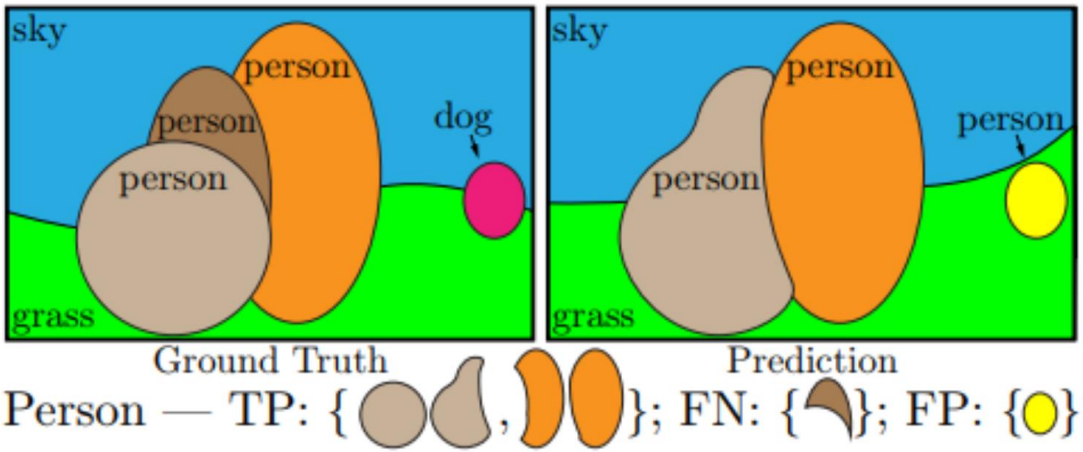
STUDENTS-HUB.com

Uploaded By: anonymous

Metrics: Panoptic quality

$$PQ = \underbrace{\frac{\sum_{(p,g)\in \mathrm{TP}} \mathrm{IoU}(p,g)}{|\mathrm{TP}|}}_{\mathrm{SQ}} \underbrace{\frac{|\mathrm{TP}|}{|\mathrm{TP}| + \frac{1}{2}|\mathrm{FP}| + \frac{1}{2}|\mathrm{FN}|}}_{\mathrm{RQ}}$$

TP = True positive, FN = False negative, FP = false positive



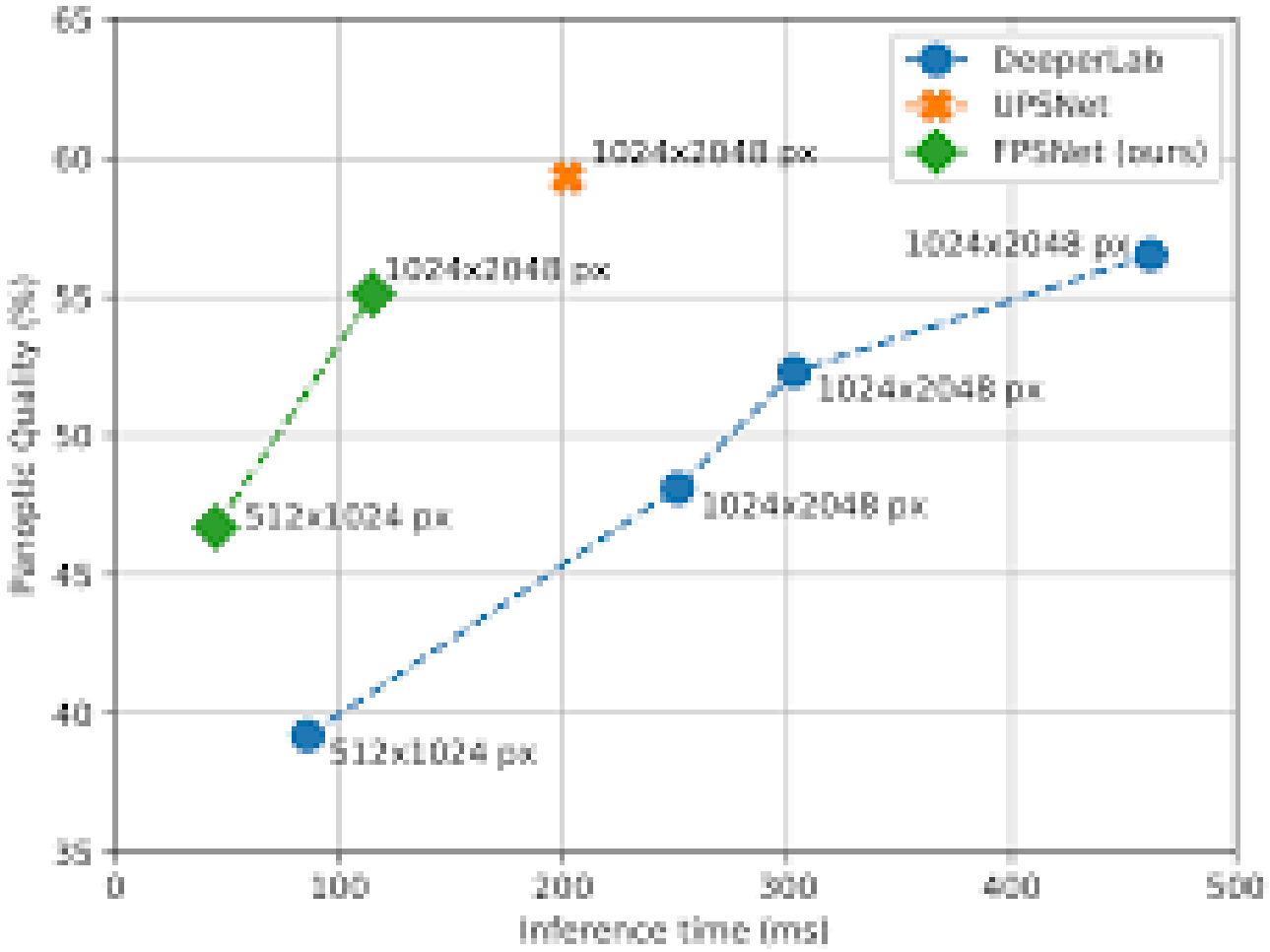
RQ: Recognition Quality = just like for detection, we want to know if we are missing any instances (FN) or we are predicting more instances (FP).

Measures the **precision and recall** of matching predicted segments with ground truth segments.

STUDENTS-HUB.com

Uploaded By: anonymous

Some Comparisons



Uploaded By: anonymous

Acknowledgement

- The material in these slides are based on:
  - Digital Image Processing: Rafael C. Gonzalez, and Richard
  - Forsythe and Ponce: Computer Vision: A Modern Approach
  - Rick Szeliski's book: Computer Vision: Algorithms and Applications
  - cs131@ Stanford University
  - cs131n@ Stanford University
  - CS198-126@ University of California, Berkely
  - CAP5415@ University of Central Florida
  - CSW182 @ University of California, Berkely
  - Deep Learning Lecture Series @UCL
  - EECS 498.008 @ University of Michigan
  - CSE576 @ Washington University
  - 11-785@ Carnegie Mellon University
  - CSCI1430@ Brown University
  - Computer Vision@ Bonn University
  - ICS 505@ KFUPM
  - Computer Vision III @ Technical University of Munich
  - Digital Image Processing@ University of Jordan

STUDENTS-HUB.com Uploaded By: anonymous