Segmentation

Alexandre Xavier Falcão

Institute of Computing - UNICAMP

afalcao@ic.unicamp.br

Introduction

Segmentation is a challenging problem which consists of two tightly coupled tasks: recognition and delineation.

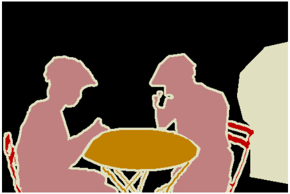
- Recognition involves detection (object's localization) and result verification (humans $\gg$ computers).
- Delineation consists of defining the precise spatial extent of the object in the image (computers $\gg$ humans).

Detection may be solved by drawing a bounding box around the object, but accurate instance and semantic delineations are very difficult with no human intervention.

Introduction

Segmentation is a challenging problem which consists of two tightly coupled tasks: recognition and delineation.

- Recognition involves detection (object's localization) and result verification (humans $\gg$ computers).
- Delineation consists of defining the precise spatial extent of the object in the image (computers $\gg$ humans).



Detection may be solved by drawing a bounding box around the object, but accurate instance and semantic delineations are very difficult with no human intervention.

Introduction

Segmentation is a challenging problem which consists of two tightly coupled tasks: recognition and delineation.

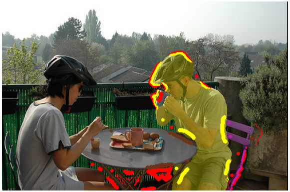
- Recognition involves detection (object's localization) and result verification (humans $\gg$ computers).
- Delineation consists of defining the precise spatial extent of the object in the image (computers $\gg$ humans).

Detection may be solved by drawing a bounding box around the object, but accurate instance and semantic delineations are very difficult with no human intervention.

Introduction

Segmentation is a challenging problem which consists of two tightly coupled tasks: recognition and delineation.

- Recognition involves detection (object's localization) and result verification (humans $\gg$ computers).
- Delineation consists of defining the precise spatial extent of the object in the image (computers $\gg$ humans).



Detection may be solved by drawing a bounding box around the object, but accurate instance and semantic delineations are very difficult with no human intervention.

Introduction

Segmentation is a challenging problem which consists of two tightly coupled tasks: recognition and delineation.

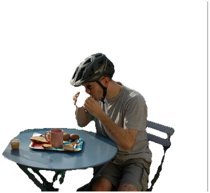
- Recognition involves detection (object's localization) and result verification (humans $\gg$ computers).
- Delineation consists of defining the precise spatial extent of the object in the image (computers $\gg$ humans).

Detection may be solved by drawing a bounding box around the object, but accurate instance and semantic delineations are very difficult with no human intervention.

Introduction

Segmentation is a challenging problem which consists of two tightly coupled tasks: recognition and delineation.

- Recognition involves detection (object's localization) and result verification (humans $\gg$ computers).
- Delineation consists of defining the precise spatial extent of the object in the image (computers $\gg$ humans).



Detection may be solved by drawing a bounding box around the object, but accurate instance and semantic delineations are very difficult with no human intervention.

Introduction

Segmentation is a challenging problem which consists of two tightly coupled tasks: recognition and delineation.

- Recognition involves detection (object's localization) and result verification (humans $\gg$ computers).
- Delineation consists of defining the precise spatial extent of the object in the image (computers $\gg$ humans).

Detection may be solved by drawing a bounding box around the object, but accurate instance and semantic delineations are very difficult with no human intervention.

Introduction

The segmentation literature is vast and it can be categorized into

- Image-based approaches, with and without shape constraints, which can be further divided into
  - pixel-based,
  - boundary-based, and
  - region-based methods.

Introduction

The segmentation literature is vast and it can be categorized into

- Image-based approaches, with and without shape constraints, which can be further divided into
  - pixel-based,
  - boundary-based, and
  - region-based methods.
- Object-based models, which learn object shape and texture (e.g., a pixel classifier) models.

Introduction

The segmentation literature is vast and it can be categorized into

- Image-based approaches, with and without shape constraints, which can be further divided into
  - pixel-based,
  - boundary-based, and
  - region-based methods.
- Object-based models, which learn object shape and texture (e.g., a pixel classifier) models.
- Hybrid techniques, which combine image- and object-based strategies.

Examples of segmentation approaches are

- pixel-based: thresholding and clustering;
- boundary-based: deformable shape models (e.g., snakes) and contour-tracking methods (e.g., live wire and riverbed);
- region-based: watershed transforms, minimum graph cut, random walks, and level sets.
- object-based: statistical shape models and fully convolutional neural networks.

- Pixel-based approaches cannot guarantee the desired number of object components.

- Pixel-based approaches cannot guarantee the desired number of object components.
- The 2D-to-3D extension of boundary-based methods is usually unnatural (more complex).

Introduction

- Pixel-based approaches cannot guarantee the desired number of object components.

- The 2D-to-3D extension of boundary-based methods is usually unnatural (more complex).

- Region-based techniques can explore optimum connectivity with seeds in 2D and 3D, but they usually require interactive seed selection.

Introduction

- Pixel-based approaches cannot guarantee the desired number of object components.

- The 2D-to-3D extension of boundary-based methods is usually unnatural (more complex).

- Region-based techniques can explore optimum connectivity with seeds in 2D and 3D, but they usually require interactive seed selection.

- Object-based models usually fail in anomolous cases.

- Pixel-based approaches cannot guarantee the desired number of object components.
- The 2D-to-3D extension of boundary-based methods is usually unnatural (more complex).
- Region-based techniques can explore optimum connectivity with seeds in 2D and 3D, but they usually require interactive seed selection.
- Object-based models usually fail in anomolous cases.

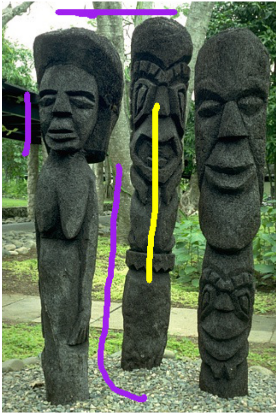
Their combination with minimal user intervention, whenever it is possible, seems to be the most reasonable alternative.

Agenda

- The role of optimum connectivity in image segmentation.
- Interactive object (instance) segmentation using the Image Foresting Transform (IFT) and FLIM [1].
- Hybrid segmentation using IFT and object-based models [2].
- Image segmentation using deep learning.

When do we need connectivity?

One can easily train a pixel classifier from image markers and select the desired object from the label map.



When do we need connectivity?

One can easily train a pixel classifier from image markers and select the desired object from the label map.

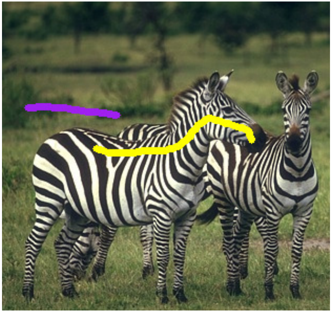
When do we need connectivity?

One can easily train a pixel classifier from image markers and select the desired object from the label map.

However, when do we need optimum connectivity?

When do we need optimum connectivity?

Optimum connectivity is needed when object and parts of the background with similar properties are connected to each other.

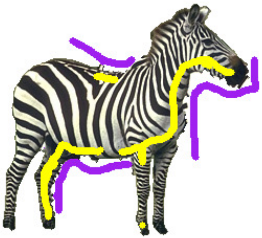


When do we need optimum connectivity?

Optimum connectivity is needed when object and parts of the background with similar properties are connected to each other.

When do we need optimum connectivity?

Optimum connectivity is needed when object and parts of the background with similar properties are connected to each other.



Such a strategy could be used to resume semantic segmentation by an optimum-path forest and correct the results in an interactive fashion [3, 4].

IFT-based object delineation

For a 2D image $\hat{I} = (D_I, \mathbf{I})$, let $(\mathcal{N}, \mathcal{A}, w)$ be such that $\mathcal{N} = D_I$, $\mathcal{A}$ is the 4-neighborhood, and w is an arc-weight function.

IFT-based object delineation

For a 2D image $\hat{I} = (D_I, \mathbf{I})$, let $(\mathcal{N}, \mathcal{A}, w)$ be such that $\mathcal{N} = D_I$, $\mathcal{A}$ is the 4-neighborhood, and w is an arc-weight function.

- Let $\mathcal{S}$ be a seed set created by internal and external markers for object delineation.

For a 2D image $\hat{I} = (D_I, \mathbf{I})$, let $(\mathcal{N}, \mathcal{A}, w)$ be such that $\mathcal{N} = D_I$, $\mathcal{A}$ is the 4-neighborhood, and w is an arc-weight function.

- Let $\mathcal{S}$ be a seed set created by internal and external markers for object delineation.
- A connectivity function suitable for object delineation is

$$
\begin{aligned}
f_{\max}(\langle p \rangle) &= \begin{cases} 0 & \text{if } p \in \mathcal{S} \\ +\infty & \text{otherwise.} \end{cases} \\
f_{\max}(\pi_p \cdot \langle p, q \rangle) &= \max\{f_{\max}(\pi_p), w_i(p, q)\},
\end{aligned}
$$

where $R(p)$ is the root of π_p, and
$w_i(p, q) = \|\mathbf{I}(R(p)) - \mathbf{I}(q)\|$ [5].

IFT-based object delineation with FLIM

For instance, FLIM can be used to compute features from image markers [1] while the object is delineated by Dynamic Trees [5].

For instance, FLIM can be used to compute features from image markers [1] while the object is delineated by Dynamic Trees [5].

IFT-based object delineation with FLIM

For instance, FLIM can be used to compute features from image markers [1] while the object is delineated by Dynamic Trees [5].

The arc weight may also be defined as

$$w(p,q) = K\left[\gamma w_o(p,q) + (1-\gamma) w_i(p,q)\right],$$

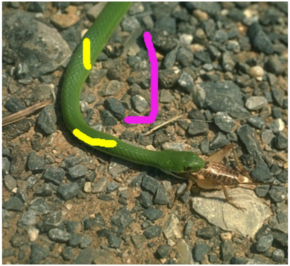
where

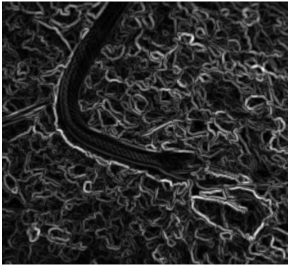
- $w_o(p,q) = |O(p) - O(q)| \in [0,1]$ is the absolute difference in object membership between p and q, according to an object map O and
- $\gamma \in [0,1]$ and $K > 1$.

IFT-based object delineation

The user can select markers and visualize the resulting arc weights as an image $\hat{D} = (D_I, D)$,

$$D(p) \quad = \quad \max_{\forall q \in \mathcal{A}(p)} \{w(p,q)\}.$$

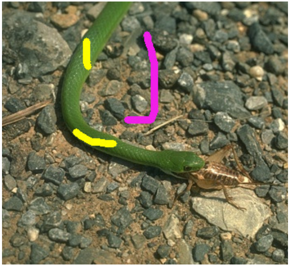


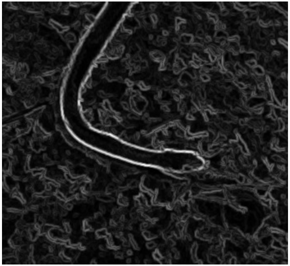


with $w_i(p,q) = \|\mathbf{I}(p) - \mathbf{I}(q)\|$ only.

The user can select markers and visualize the resulting arc weights as an image $\hat{D} = (D_I, D)$,

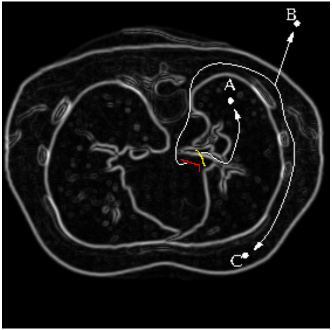
$$D(p) = \max_{\forall q \in \mathcal{A}(p)} \{w(p,q)\}.$$

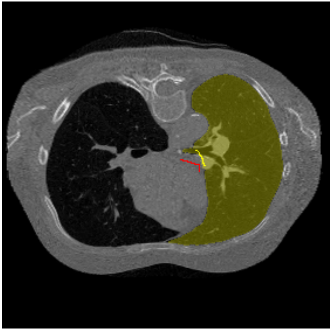




with $w_o(p,q)$ and $w_i(p,q)$, for $\gamma = 0.8$.

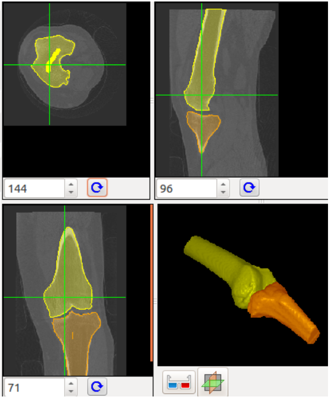
Seeds must be selected around low-contrast parts of the object's boundary.

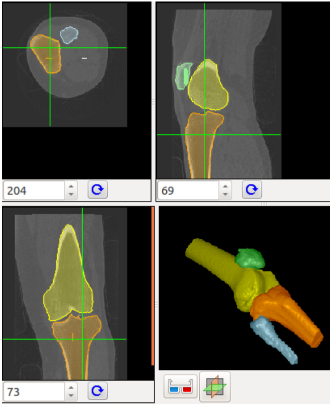




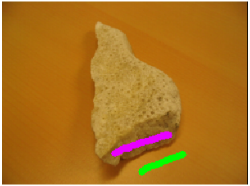
Seed addition/removal for delineation

The segmentation can be edited by adding and/or removing seeds, such that the forest is reconstructed in a differential way [6].

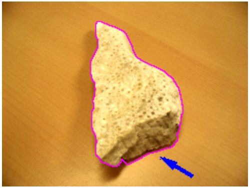




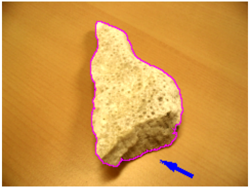
Region-based delineation using border orientation



original

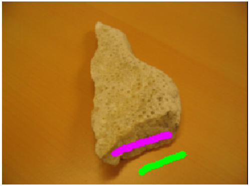


non-oriented

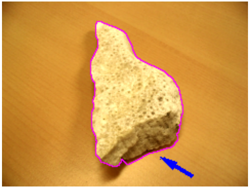


oriented

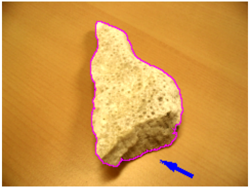
Region-based delineation using border orientation



original



non-oriented



oriented

$$
\begin{aligned}
f_{omx}(\langle p \rangle) &= \begin{cases} 0 & \text{if } p \in \mathcal{S}_i \cup \mathcal{S}_e \\ +\infty & \text{otherwise,} \end{cases} \\
f_{omx}(\pi_p \cdot \langle p, q \rangle) &= \max\{f_{omx}(\pi_p), w(p,q)\} \\
w(p,q) &= \begin{cases} w_i^{1.5}(p,q) & \text{if } O(p) > O(q) \text{ and } R(p) \in \mathcal{S}_i, \\ w_i^{1.5}(p,q) & \text{if } O(p) < O(q) \text{ and } R(p) \in \mathcal{S}_e, \\ w_i(p,q) & \text{otherwise,} \end{cases}
\end{aligned}
$$

where $\mathcal{S}_i$ and $\mathcal{S}_e$ are internal and external seeds, and $R(p)$ is the root of p [7].

The IFT algorithm with f_{omx} can be described as follows.

Input: Arc-weighted graph $(D_I, \mathcal{A}, w_i')$ and seed set $\mathcal{S}_i \cup \mathcal{S}_e$ labeled by function λ: $\mathcal{S}_i \rightarrow \{1\}$ and $\mathcal{S}_e \rightarrow \{0\}$.

Output: Label image $\hat{L} = (D_I, L)$.

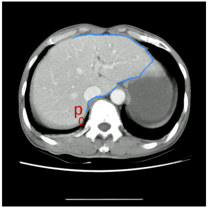
Auxiliary: Priority queue $\mathcal{Q}$, connectivity map V, root map R, and var

Instead of using $R(p) \in \mathcal{S}_i$ and $R(p) \in \mathcal{S}_e$, one may use $L(p) = 1$ and $L(p) = 0$, respectively, when computing $w(p, q)$.

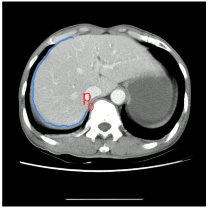
Region-based delineation using border orientation

1. For each $p \in D_I$, do.
2. If $p \in \mathcal{S}_i \cup \mathcal{S}_e$ then.
3. Set $L(p) \leftarrow \lambda(p)$, $R(p) \leftarrow p$ and $V(p) \leftarrow 0$.
4. Else, set $V(p) \leftarrow +\infty$.
5. Insert p in $\mathcal{Q}$.
6. While $\mathcal{Q} \neq \emptyset$, do.
7. Remove p from $\mathcal{Q}$ such that $p = \arg\min_{\forall q \in \mathcal{Q}}\{V(q)\}$.
8. For each $q \in \mathcal{A}(p) \mid q \in \mathcal{Q}$, do.
9. Set $tmp \leftarrow \max\{V(p), w(p,q)\}$.
10. If $tmp < V(q)$, then.
11. Set $V(q) \leftarrow tmp$, $R(q) \leftarrow R(p)$ and $L(q) \leftarrow L(p)$.
12. Return $\hat{L} = (D_I, L)$.

Oriented boundary delineation

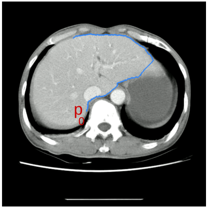


non-oriented

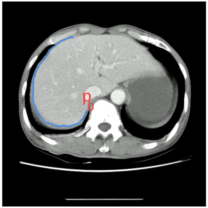


clockwise oriented

Oriented boundary delineation



non-oriented



clockwise oriented

$$
\begin{aligned}
f_{lwr}(\langle p \rangle) &= \begin{cases} 0 & \text{if } p = p_0 \\ +\infty & \text{otherwise,} \end{cases} \\
f_{lwr}(\pi_p \cdot \langle p, q \rangle) &= f_{lwr}(\pi_p) + w(p,q) \\
w(p,q) &= \begin{cases} [K - w_i(r,l)]^{1.5} & \text{if } O(l) > O(r), \\ K - w_i(r,l) & \text{otherwise,} \end{cases}
\end{aligned}
$$

where p_0 is an initial point on the boundary, r and l are the right (interior) and left (exterior) pixels of (p, q) [8].

Oriented boundary delineation from p_0 to p_1

The IFT algorithm with f_{lwr} can be described as follows.

Input: Arc-weighted graph $(D_I, \mathcal{A}, w_i')$, seed pixel p_0, and target pixel p_1.

Output: Predecessor image $\hat{P} = (D_I, P)$.

Auxiliary: Priority queue $\mathcal{Q}$, connectivity map V, and variable tmp.

Oriented boundary delineation from p_0 to p_1

1. For each $p \in D_I$, do.
2. If $p = p_0$ then.
3. Set $P(p) \leftarrow nil$ and $V(p) \leftarrow 0$.
4. Else, set $V(p) \leftarrow +\infty$.
5. Insert p in $\mathcal{Q}$.
6. While $\mathcal{Q} \neq \emptyset$, do.
7. Remove p from $\mathcal{Q}$ such that $p = \arg\min_{\forall q \in \mathcal{Q}}\{V(q)\}$.
8. If $p = p_1$ then return $\hat{P} = (D_I, P)$.
9. For each $q \in \mathcal{A}(p) \mid V(q) > V(p)$, do.
10. Set $tmp \leftarrow V(p) + w(p, q)$.
11. If $tmp < V(q)$, then.
12. Set $V(q) \leftarrow tmp$ and $P(q) \leftarrow p$.

- The path $\pi = \pi_{p_0 \rightarrow p_1}$ is obtained by calling $Path(P, \pi, p_0, p_1)$ for π initially empty.

  $Path(P, \pi, p_0, p)$:
  1. If $(p = p_0)$ then.
  2. Return $\pi \leftarrow \langle p_0 \rangle$.
  3. Else
  4. Return $\pi \leftarrow Path(P, \pi, p_0, P(p)) \cdot \langle P(p), p \rangle$.

- Now suppose p_1 is the cursor position and it moves to p_2, the predecessor map P can be updated by the following incremental algorithm.

Oriented boundary delineation from p_0 to p_2

1. If $p_2 \notin \mathcal{Q}$ then return $\hat{P} = (D_I, P)$.
2. While $\mathcal{Q} \neq \emptyset$, do.
3. Remove p from $\mathcal{Q}$ such that $p = \arg\min_{\forall q \in \mathcal{Q}} \{V(q)\}$.
4. If $p = p_2$ then return $\hat{P} = (D_I, P)$.
5. For each $q \in \mathcal{A}(p) \mid V(q) > V(p)$, do.
6. Set $tmp \leftarrow V(p) + w(p, q)$.
7. If $tmp < V(q)$, then.
8. Set $V(q) \leftarrow tmp$ and $P(q) \leftarrow p$.

How can texture and shape models substitute a human in object localization?

Hybrid segmentation using IFT and object models

How can texture and shape models substitute a human in object localization?

- A shape model estimates markers, while it translates and imposes shape constraints for object delineation.

How can texture and shape models substitute a human in object localization?

- A shape model estimates markers, while it translates and imposes shape constraints for object delineation.
- A criterion function selects the object at the location of maximum delineation score.

How can texture and shape models substitute a human in object localization?

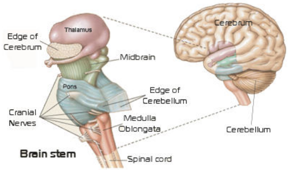
- A shape model estimates markers, while it translates and imposes shape constraints for object delineation.
- A criterion function selects the object at the location of maximum delineation score.

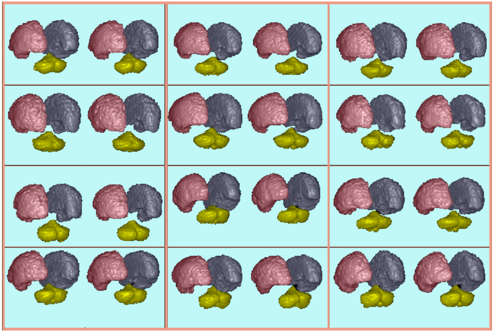
How can texture and shape models substitute a human in object localization?

- A shape model estimates markers, while it translates and imposes shape constraints for object delineation.
- A criterion function selects the object at the location of maximum delineation score.

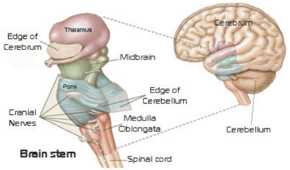
A voxel classifier can reduce candidate locations and avoid wrong marker selection.

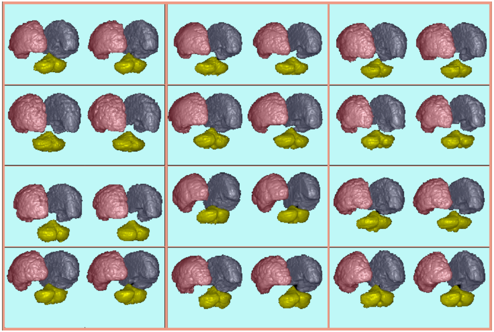
Result of model-based segmentation of the brain hemispheres and cerebellum without the brain stem.





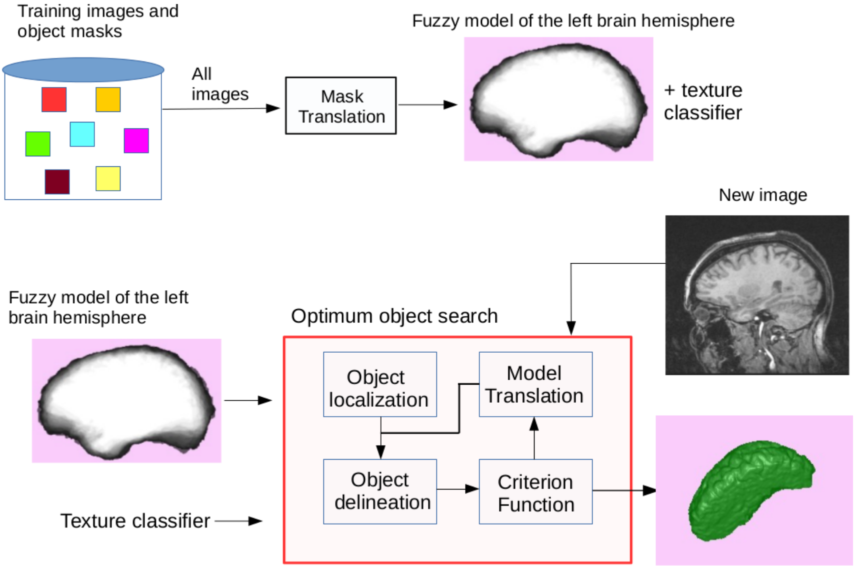
Result of model-based segmentation of the brain hemispheres and cerebellum without the brain stem.



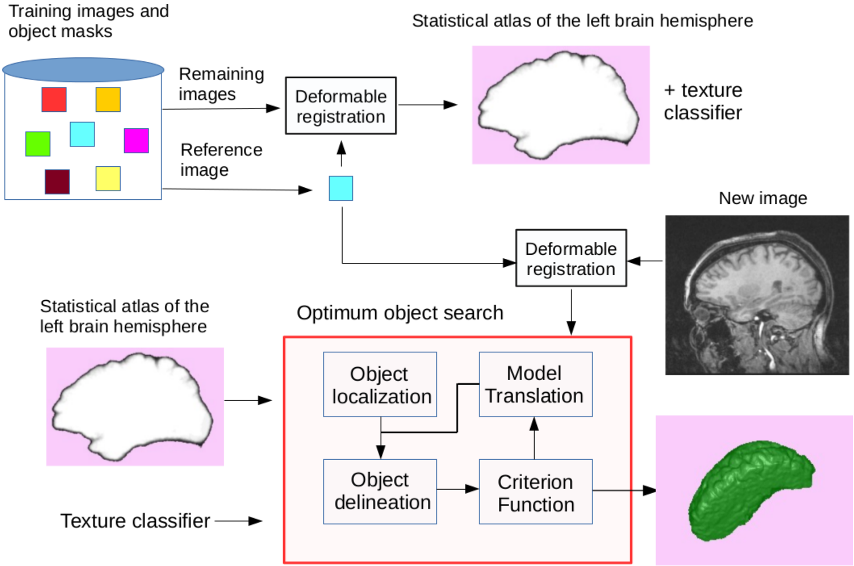


We have developed two types of shape models under this segmentation paradigm, fuzzy models [9] and statistical atlases [10, 11, 2].

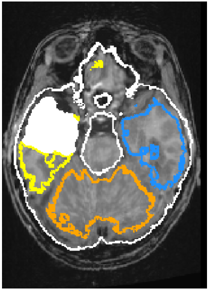
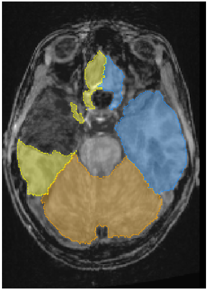
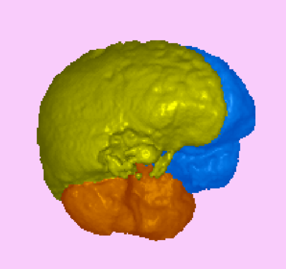
Fuzzy object shape models: construction and use



Statistical atlases: construction and use



Abnormal brain: the voxel classifier avoids internal markers in surgically removed regions of the brain.

- Semantic segmentation has been addressed with no user intervention.

- Semantic segmentation has been addressed with no user intervention.
- Successful methods start by using fully convolutional neural networks – i.e., rather than applying CNNs to patches around pixels, convolutional layers extract image features and a pixel classifier predicts their labels [12].

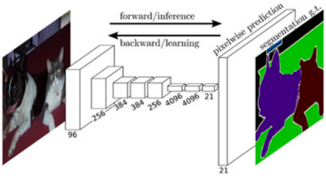


Figure from [12].

- Subsequenty, the predictor was substituted by deconvolutional layers (convolution with upsampling) forming an encoder-decoder architecture, with skip connections [13].

Image segmentation with deep learning

- Subsequenty, the predictor was substituted by deconvolutional layers (convolution with upsampling) forming an encoder-decoder architecture, with skip connections [13].

- However, as downsampling used by pooling is crucial for object detection, accurate boundary delineation becomes compromised.

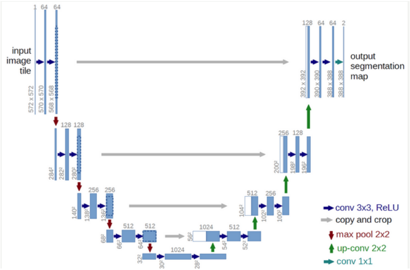


Figure from [13].

Atrous (dilated) convolution and spatial pyramidal pooling have been introduced and combined to alleviate the downsizing problem and capture multiscale information [14].

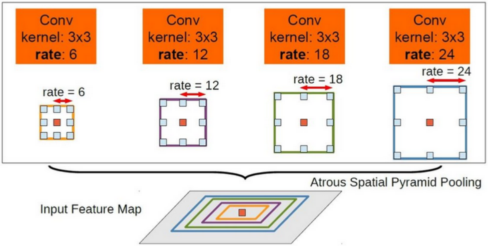


Figure from [14].

Atrous (dilated) convolution and spatial pyramidal pooling have been introduced and combined to alleviate the downsizing problem and capture multiscale information [14].

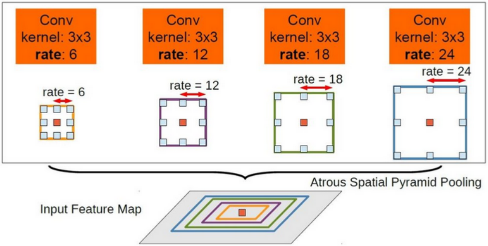


Figure from [14].

However, boundary delineation is still compromised since the backbone comes from CNNs (VGG-16 or ResNet-101) used for image classification.

Image segmentation with deep learning

A compromise between detection and delineation is still subject of investigation by combining large kernels for localization (Global Convolutional Blocks), boundary refinement blocks [15, 16], and kernel-sharing atrous convolution (when dilated kernels are combined into a single one) [17].

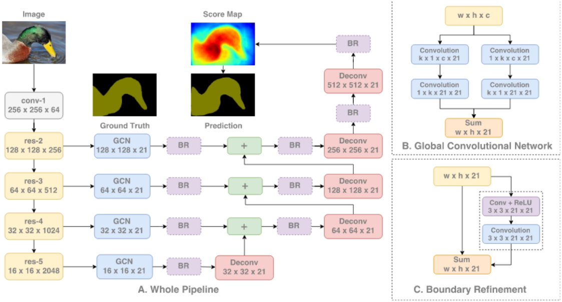


Figure from [15].

Still, user interaction is needed to improve prediction, especially for instance segmentation, and with no guarantee that the classified pixels are connected to the user clicks [18].

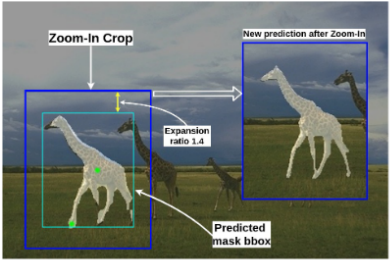


Figure from [18].

Image segmentation with deep learning

Still, user interaction is needed to improve prediction, especially for instance segmentation, and with no guarantee that the classified pixels are connected to the user clicks [18].

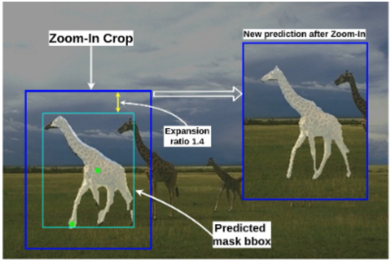


Figure from [18].

Could we use the IFT to add connectivity information and improve object delineation in deep learning?

[1] I. E. de Souza, B. C. Benato, and A. X. Falcão.

Feature learning from image markers for object delineation.

In *2020 33rd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 116–123, 2020.

doi: 10.1109/SIBGRAPI51738.2020.00024.

[2] S.B. Martins, J. Bragantini, A.X. Falcão, and C.L. Yasuda.

An adaptive probabilistic atlas for anomalous brain segmentation in mr images.

Medical Physics, 46(11):4940–4950, 2019.

[3] Anderson Carlos Moreira Tavares, Paulo André Vechiatto Miranda, Thiago Vallin Spina, and Alexandre Xavier Falcão.

A supervoxel-based solution to resume segmentation for interactive correction by differential image-foresting transforms.

In Jesús Angulo, Santiago Velasco-Forero, and Fernand Meyer, editors, *Mathematical Morphology and Its Applications to Signal and Image Processing*, pages 107–118, Cham, 2017. Springer International Publishing.

[4] P. A. V. Miranda, A. X. Falcão, G. C. S. Ruppert, and F. A. M. Cappabianco.
How to fix any 3d segmentation interactively via image foresting transform and its use in mri brain segmentation.
In *2011 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 2031–2035, 2011.

[5] Jordão Bragantini, Samuel Botter Martins, Cesar Castelo-Fernandez, and Alexandre Xavier Falcão.
Graph-based image segmentation using dynamic trees.
In *Iberoamerican Congress on Pattern Recognition*, pages 470–478, 2018.

[6] A. X. Falcão and F. P. G. Bergo.
Interactive volume segmentation with differential image foresting transforms.
IEEE Transactions on Medical Imaging, 23(9):1100–1108, Sep 2004.

[7] P.A.V. Miranda and L.A.C. Mansilla.
Oriented image foresting transform segmentation by seed competition.

IEEE Transactions on Image Processing, 23(1):389–398, Jan 2014.

[8] A.X. Falcão, J.K. Udupa, S. Samarasekera, S. Sharma, B.E. Hirsch, and R. de A. Lotufo.

User-steered image segmentation paradigms: Live wire and live lane.

Graphical Models and Image Processing, 60(4):233 – 260, 1998.

[9] P.A.V. Miranda, A.X. Falcão, and J.K. Udupa.

Cloud Models: Their Construction and Employment in Automatic MRI Segmentation of the Brain.

Technical Report IC-10-08, Institute of Computing, University of Campinas, March 2010.

In English, 18 pages.

[10] R. Phellan, A.X. Falcão, and J.K. Udupa.

Medical image segmentation via atlases and fuzzy object models: Improving efficacy through optimum object search and fewer models.

Medical Physics, 43(1):401–410, 2016.

[11] S.B. Martins, T.V. Spina, C. Yasuda, and A.X. Falcão.

A multi-object statistical atlas adaptive for deformable registration errors in anomalous medical image segmentation.

In *Proc.SPIE*, volume 10133, pages 10133 – 10133 – 8, 2017.

[12] E. Shelhamer, J. Long, and T. Darrell.

Fully convolutional networks for semantic segmentation.

IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(4):640–651, 2017.

[13] O. Ronneberger, P.Fischer, and T. Brox.

U-net: Convolutional networks for biomedical image segmentation.

In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 9351 of *LNCS*, pages 234–241. Springer, 2015.

(available on arXiv:1505.04597 [cs.CV]).

[14] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille.

Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs.

IEEE Transactions on Pattern Analysis and Machine Intelligence, 40(4):834–848, 2018.

[15] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun.
Large kernel matters — improve semantic segmentation by global convolutional network.
In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1743–1751, 2017.

[16] Chuan Kai Wang. and Long Wen Chang.
Semantic segmentation via global convolutional network and concatenated feature maps.
In *Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods - Volume 1: ICPRAM,*, pages 292–297. INSTICC, SciTePress, 2019.

[17] Ye Huang, Qingqing Wang, Wenjing Jia, and Xiangjian He.
See more than once – kernel-sharing atrous convolution for semantic segmentation, 2019.

[18] K. Sofiiuk, I. Petrov, O. Barinova, and A. Konushin.

F-brs: Rethinking backpropagating refinement for interactive segmentation.

In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8620–8629, 2020.