Summaries of a few papers on mitral valve prolapse and cross-modal image registration

Stephan

March 5, 2020

Contents

1 Image registration

1.1 3d Point set registration

1.1.1 Point set to point set: Iterative Closest Point algorithm [Besl and McKay, 1992, Chen and Medioni, 1991]

An algorithm to register two sets of points. The basic principle is that it is easy to register two sets of points if the points are matched in pairs, and hard if they are not. See Algorithm 1.

Algorithm 1 Iterative Closest Point

```
Input: two sets of points $X$ and $Y$; “find closest point” procedure; “find best transformation” procedure
Output: a transformation from $Y$ to $X$
  repeat
    for every point $y \in Y$ do
      find the point $x(y) \in X$ that is closest to $y$
    end for
    find the transformation $q$ that best maps all points in $Y$ to their image in $X$
    apply $q$ to $Y$
  until $\sum_{y \in Y} ||q(y) - y|| \leq$ some threshold, or number of iterations $\geq$ some threshold
  return the transformation obtained by composing all the successive transformations
```

1.1.2 Model to point set: Random Sample Consensus (RanSaC) algorithm [Fischler and Bolles, 1981]

The RanSaC algorithm: fit a model to a set of points. Assumes the model can be fitted using a small number of points, but can also fit “as best as possible” a larger number of points, for instance using least squares. The algorithm is robust to outliers. The algorithm is pretty general, and can be further adapted to a specific setting. It is random, but could possibly be made deterministic. See Algorithm 2.

Algorithm 2 Random Sample Consensus

```
Input: a set of $N$ points; a model that requires $n < N$ points to instantiate; a tolerance threshold $\alpha$; a number of points threshold $t$
Output: an instance of the model
  loop
    select $n$ points at random
    instantiate the model using these $n$ points
    find all points that fit the model within a tolerance error $\leq \alpha$
    if the number of points is at least $t$ then
      use all these points to instantiate a new model
      return the new model
    end if
  end loop
  if the loop ended without returning a model then
    return failure; or alternatively, the model that fitted the highest number of points within $\alpha$
  end if
```

1.1.3 Cross-modal point cloud registration [Huang et al., 2016]

Proposes an algorithm for registration of 3d clouds of points, with more than a thousand points: possibly tens of thousands or millions of points. The two clouds may have been generated using different kinds of sensors, and thus present differences, such as inequal point density or missing parts

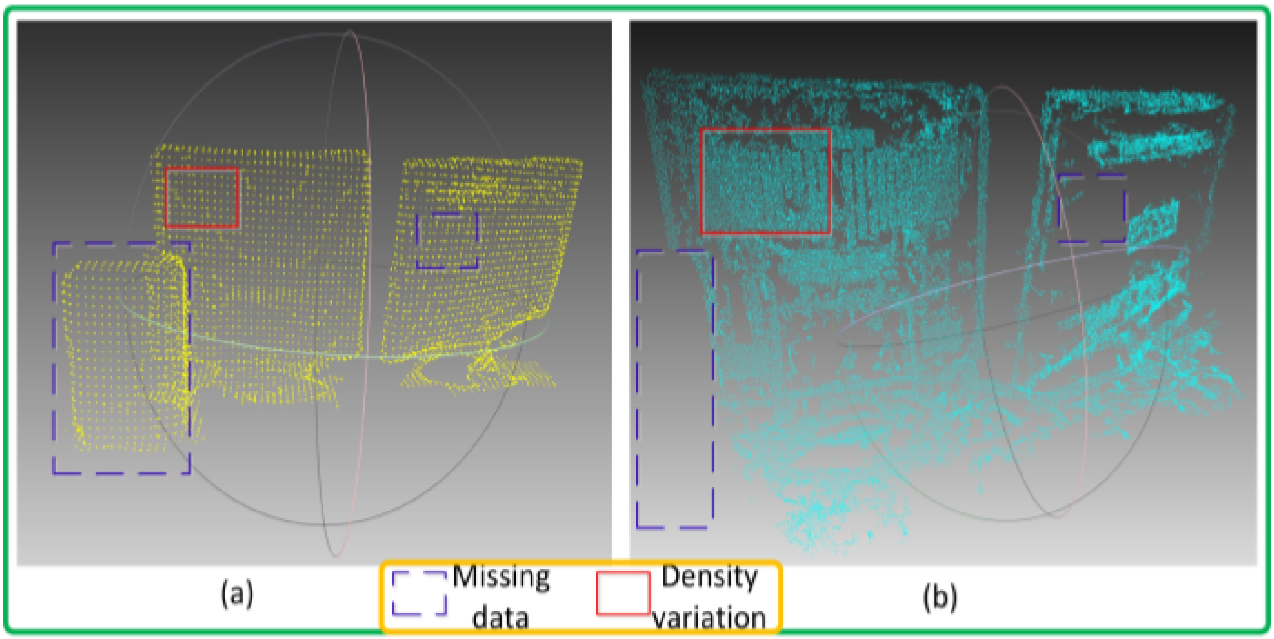

Figure 1: Difference in point density between Kinect (a) and a cloud reconstructed from multiple 2d RGB cameras (b). Illustration from [Huang et al., 2016].

of the represented object: see Fig. 1. The idea behind the algorithm is to extract the "macro" "micro" structures from the images, and use them to build a graph for each image representing those structures. The point cloud registration problem becomes a graph matching problem. The macro structure extraction algorithm is supervoxel clustering [Papon et al., 2013]. The micro structure extraction consists in adding features on the vertices of the graph; these features are Ensemble Shape Functions [Wohlkinger and Vincze, 2011] of each cluster. See Algorithm 3.

Mostly focused on point clouds obtained from these two sources: 1) Kinect; 2) reconstruction from multiple 2D RGB cameras.

```
Algorithm 3 Point Cloud Registration [Huang et al., 2016]
Input: two 3d point clouds
Output: registration of the two clouds
  normalize the scales of the two images
  supervoxel clustering: build a graph for each point cloud
  add features on each vertex by computed the ensemble shape function of its cluster of points
  find best matching between the two graphs
  remove mismatched vertices (”outliers”) using ICP/RanSaC
  deduce the 3d transformation corresponding to the graph matching
  return the 3d transformation
```

The micro and macro structure extraction consists in clustering voxels (3d pixels) into supervoxels. Look for the sourcecode of the clustering algorithm in the Point Cloud Library [Rusu and Cousins, 2011]. To solve the graph matching problem, see [Zhou and De la Torre, 2013, Cour et al., 2007, Zaslavskiy et al., 2008, Frank and Wolfe, 1956, Gertz and Wright, 2003].

1.1.4 Point cloud registration with outliers [Papazov and Burschka, 2011]

Note that ICP amounts to searching for a rigid transformation T which minimizes the sum of distances between closest points, for two pointclouds D and M:

$$\sum_{x\in D} d(T(x), M)$$

Propose to search instead for a rigid transformation T by minimising instead the sum:

$$\sum_{x \in D} -\varphi(d(T(x), M))$$

where $-\varphi$ is a function which is both increasing and bounded, satisfying $-\varphi(0) = -1$ and $\lim_{d \to +\infty} -\varphi(d) = 0$. The function $-\varphi$ becomes almost constant for large values of the distance d; hence points that are very badly matched cannot have a strong incidence on the minimisation. In a way, the algorithm proposed in this paper is a variant of ICP robust to outliers. It could also be used to detect and remove outliers?

The used formula for φ is:

$$\varphi(d) = \frac{1}{1 + \alpha d}$$

for some parameter α.

The parameter α can be chosen as a function of two parameters d_0 and δ so that $\varphi(d) < \delta$ as soon as $d > d_0$, using the formula:

$$\alpha = \frac{1 - \delta}{\delta {d_0}^2}$$

In their implementation, they choose $\delta = 0.1$ and set d_0 to 1/4 of the minimum of the three dimensions of the bouding box of the point cloud.

This paper also gives very cool formulae to perform dichotomy on the space of 3d rotations, and to select ”uniformly at random” rotations in this space or in the dichotomy subspaces.

1.2 Intensity-based registration

1.2.1 Using intensity for the similarity measure

Need to find more references about this.

Only useful if all images come from the same modality. Usually requires intensity normalization of the two images.

1.2.2 Electron microscopy to light microscopy [Toledo Acosta et al., 2018]

Intensity-based registration, specific for 3d light microscopy (LM) to 3d electron microscopy (EM). The 3d images are stacks; in particular, the size of a pixel in the z-axis differs from the size in the x and y axes.

First, a translation is computed to pre-align the images in the xy-plane, because the misalignment in that plane is expected to be greater than in the z-axis. A 3d region of interest is selected in the stack, by human intervention or by some other algorithm. The light microscopy stack is projected into a 2d image using maximum-intensity projection, while a few slices are selected from the electron microscopy stack. Then each EM slice is aligned to the LM projection, using a rotation-invariant similiarity measure on a histogram-based description of the Laplacian-of-Gaussian of each of the two images. The full LM stack is shifted by a weighted average of the translations corresponding to each EM slice.

Second, a 2d affine transformation is computed to align the 3d LoG-EM-RoI and the pre-aligned 3d LoG-LM-RoI. The LoG-LM stack is resampled with a low-pass filter and a bilinear interpolator, so that pixel size for the two images are the same. The transformation is computed using the similarity measure Mutual Information, and is computed in two steps: first, a rigid transformation, ie, a direct isometry; then, an additional affine transformation.

1.3 Cross-modal registration using image analogies

1.3.1 Generating an image looking like modality B given an image coming from modality A [Hertzmann et al., 2001]

See Fig. 2.

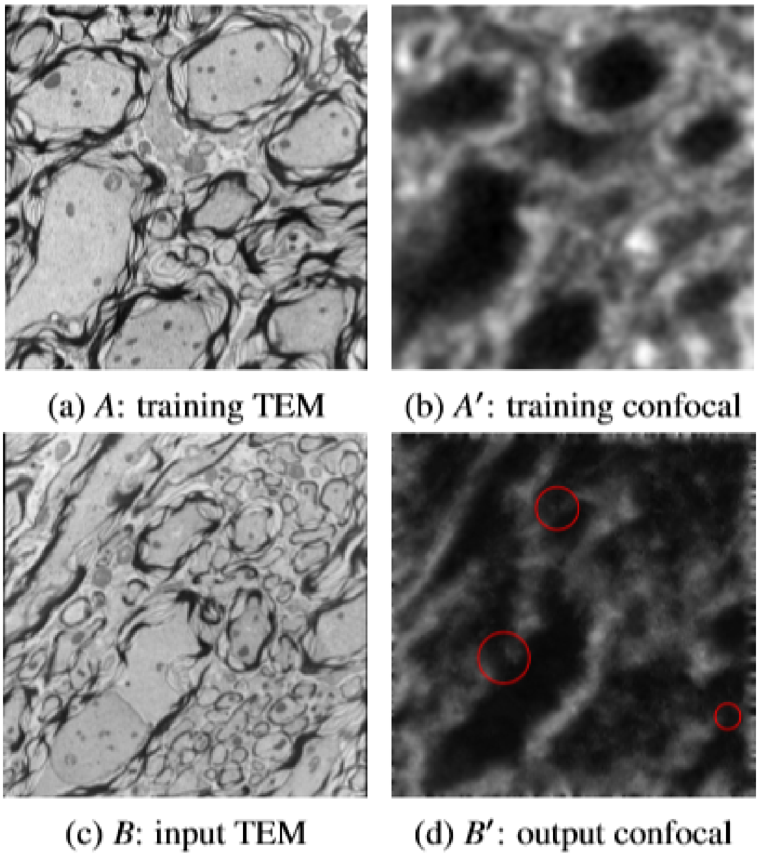

Figure 2: After training on pairs of images coming from two different modalities, the neural network can simulate an image from the second modality when given an image from the first modality. Illustration from [Cao et al., 2014].

1.3.2 Building dictionaries of features [Cao et al., 2014]

Uses machine learning on a training dataset of pairs of images from two different modalities to learn two dictionaries D_A and D_B of small filters for each modality.

To register an image I_A from modality A with an image I_B from modality B: first, the image I_A from modality A is converted into a sparse vector α such that I_A can be reconstructed from vector α and dictionary D_A. Then an image I'_B is constructed from α and D_B. Then images I_B and I'_B can be registered using a uni-modality registration algorithm. See Fig. 3.

1.3.3 How to solve the registration problem with the reconstructed image

1. Denoising the image rebuilt from the dictionary of features [Elad and Aharon, 2006, Li and Liu, 2009] (note: surprisingly, [Cao et al., 2014] does not cite [Li and Liu, 2009]) and denoising the target image

2. "We consider rigid followed by affine and B-spline registrations in this paper and use elastix's implementation [Klein et al., 2010, Johnson et al., 2015]. As similarity measures we use sum of squared differences (SSD) and mutual information (MI). A standard gradient descent is used for optimization. For B-spline registration, we use displacement magnitude regularization which penalizes $||T(x) - x||^2$, where $T(x)$ is the transformation of coordinate x in an image [Klein et al., 2010]. This is justified as we do not expect large deformations between the images as they represent the same structure. Hence, small displacements are expected, which are favored by this form of regularization." [Cao et al., 2014]

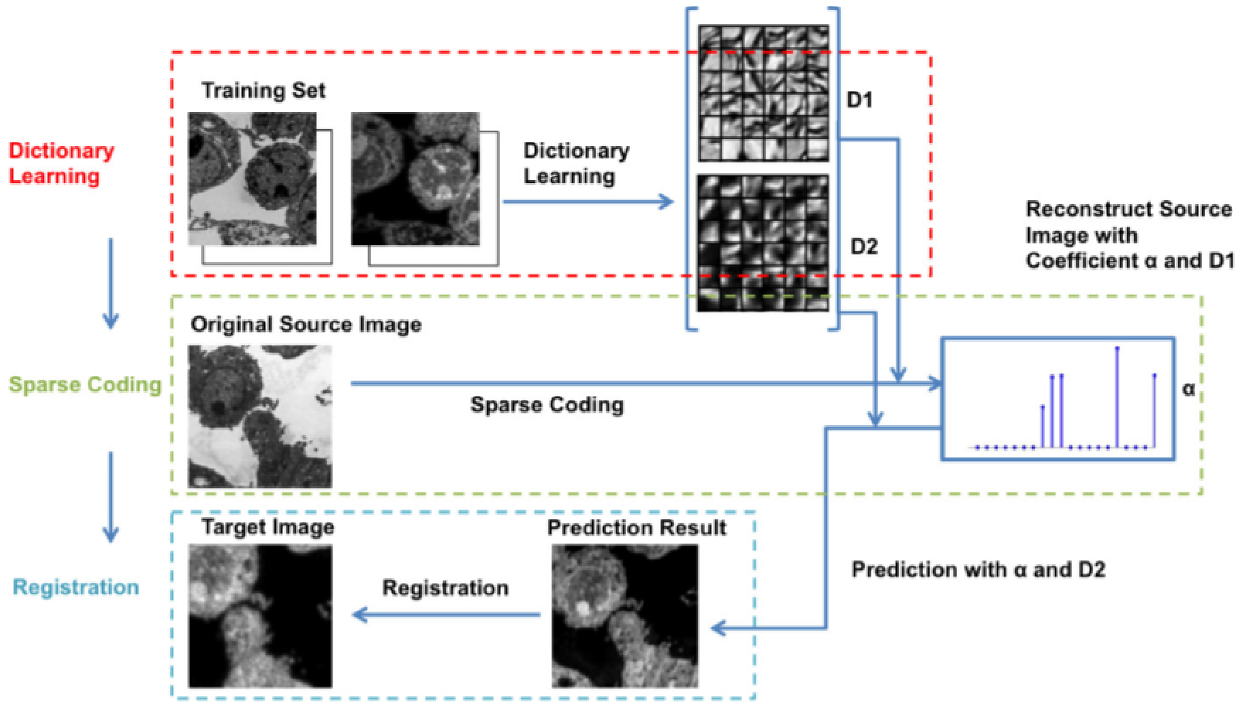

Figure 3: Summary of the registration algorithm using deep learning to extract dictionaries of features. Illustration from [Cao et al., 2014].

1.4 Using similarity measures with deep learning [Litjens et al., 2017]

1.4.1 Extracting features in MR brain images with unsupervised deep learning [Wu et al., 2013]

Proposes an unsupervised deep learning approach to directly learn the basis filters that can effectively represent all observed image patches in magnetic resonance (MR) brain images. The coefficients by these learnt basis filters in representing the particular image patch can be regarded as the morphological signature for correspondence detection during image registration. Specifically, a stacked two-layer convolutional network is constructed to seek for the hierarchical representations for each image patch, where the high-level features are inferred from the responses of the low-level network.

Not multimodal: only MR images. But the training is unsupervised.

1.4.2 Using a convolutional neural network to compute the similarity measure in multi-modal registration [Simonovsky et al., 2016]

Proposes a metric based on a convolutional neural network, to replace Mutual Information as a similarity measure. The network can be trained from scratch even from a few aligned image pairs. The metric is validated on intersubject deformable registration.

Supervised, but requires “only a few” aligned image pairs. Multi-modal?

1.4.3 Using stacked auto-encoders to compute the similarity measure in multi-modal registration [Cheng et al., 2018]

Trains a binary classifier to learn the correspondence of two image patches. The classification output is transformed to a continuous probability value, then used as the similarity score. Proposes to utilise multi-modal stacked denoising autoencoder to effectively pre-train the deep neural network. Train and test the proposed metric using sampled corresponding/non-corresponding computed tomography and magnetic resonance head image patches from a same subject.

Multi-modal: MR and CT.

2 Image segmentation

2.1 Cell segmentation

See Fig. 4.

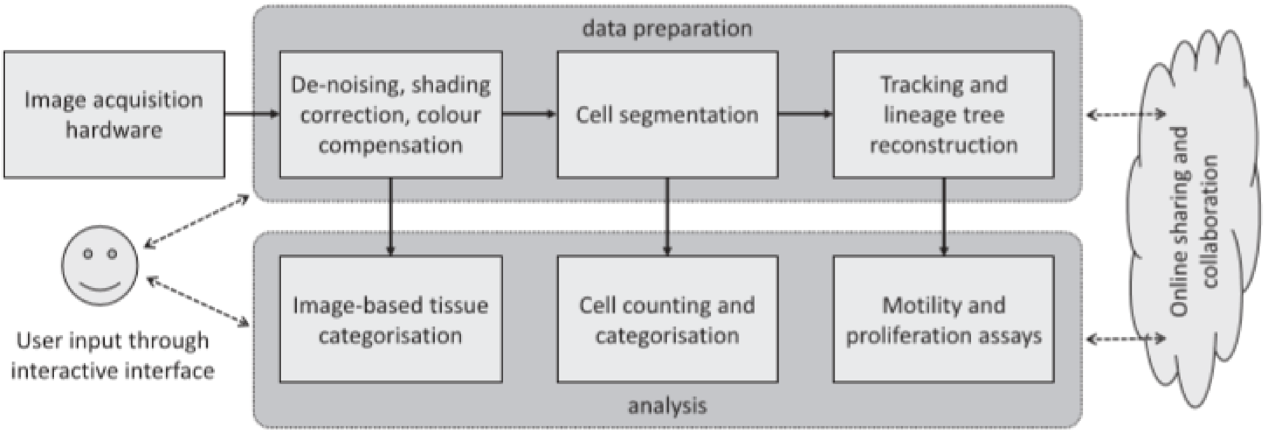
data preparation
Image acquisition hardware
De-noising, shading correction, colour compensation
Cell segmentation
Tracking and lineage tree reconstruction
Online sharing and collaboration
User input through interactive interface
Image-based tissue categorisation
Cell counting and categorisation
Motility and proliferation assays
analysis

Figure 4: Common tasks in analysis of cell images. Illustration from [Kan, 2017].

2.1.1 Colour threshold and pixel adjacency

Apply filters to remove noise, blur, and non-cell objects. Use an intensity threshold to classify pixels as part of a cell or part of the background. Use connected component analysis to separate non-touching cells. If cells overlap, use a "watershed" technique to separate them.

2.1.2 After segmentation, decision tree to classify an image segment as a cell or not a cell based on its area and elongation [Kan, 2017]

After segmenting the objects in the image, classify each object as a cell or not a cell. Possible features for the classifier:

- area (small objects are not cells);
- elongation $4\pi A/P^2$, where A is area and P is perimeter (close to 1 if object is shaped like a disc, close to 0 if it is shaped like a line).

2.1.3 Convolutional neural networks

2018 Data Science Bowl [Marks, 2018] A contest on kaggle to devise algorithms for cell segmentation. The algorithms are required to be able to handle images from several different experiments using light microscopy. The three best ranked submissions to the contest use convolutional neural networks and perform significantly better than the software CellProfiler, while not requiring the user to spend time configuring the software. The good performance of these solutions appears to come in particular from the succesful use of data augmentation and data preprocessing. See Fig. 5.

2.1.4 Tracking and constructing lineage trees

Cells are observed over time. Need to track individual cells over time. Cells might overlap in some frames; a cell might divide through mitosis.

Classification problem over two successive segmented frames Given two successive frames, classify all pairs (cell from first frame, cell from second frame) as "same cell" or "distinct cells" [Kan, 2017]. See Fig. 6.

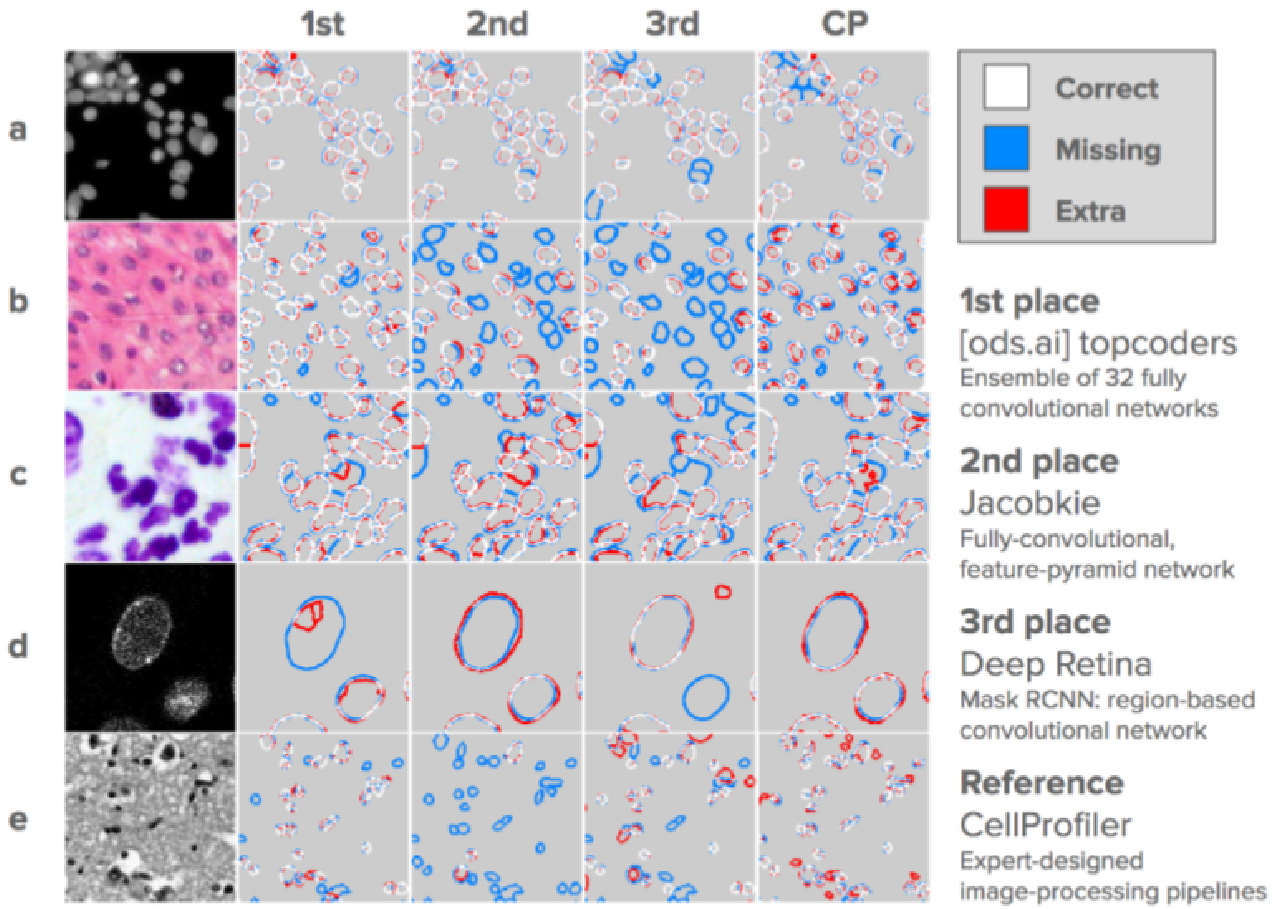
1st
2nd
3rd
CP
a
b
c
d
e
Correct
Missing
Extra
1st place
[ods.ai] topcoders
Ensemble of 32 fully convolutional networks
2nd place
Jacobkie
Fully-convolutional, feature-pyramid network
3rd place
Deep Retina
Mask RCNN: region-based convolutional network
Reference
CellProfiler
Expert-designed image-processing pipelines

Figure 5: Illustration from [Marks, 2018].

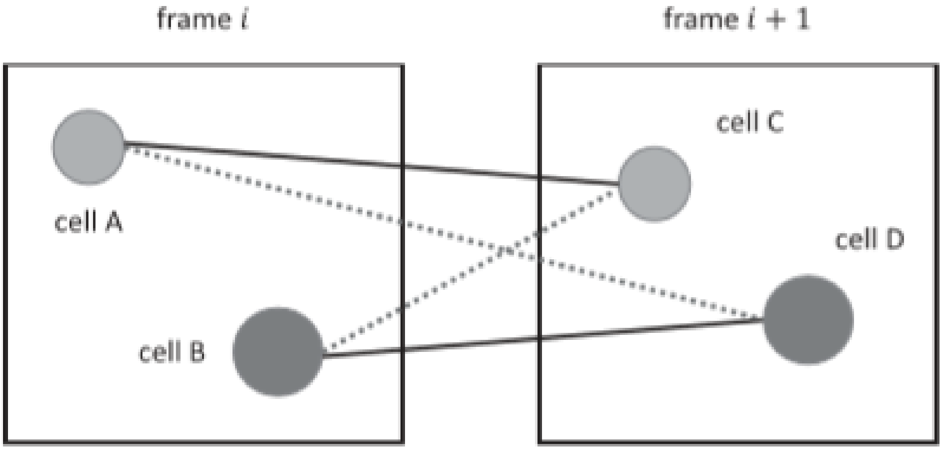

Figure 6: Illustration from [Kan, 2017].

2.2 Pixel classification

2.2.1 IlastiK [Berg et al., 2019]

A program in python which implements supervised pixel classification, as well as a few other segmentation problems. The algorithm for pixel classification in IlastiK is described in Algorithm 4. The program provides a paintbrush-like interface to ”paint” some pixels to mark them with a training label. 3d images can be viewed by slices along three axes. There is a two-step variant of pixel classification based on Tu and Bai’s autocontext algorithm [Tu and Bai, 2010]. IlastiK also provides a few other functionalities, such as object classification, or tracking of moving segmented objects across frames.

Algorithm 4 IlastiK Pixel Classification [Berg et al., 2019]

```
Input: an image or a sequence of images
Output: classification of each pixel in the images
  Choose a set of predefined convolutional filters, such as edge detection or Laplacian of Gaussian
  Compute features for every pixel
  Ask the user to ”paint” some pixels with a label, using a paintbrush
  Build a random forest using the computed features for each pixel and the user-provided labels
  Apply the random forest to every pixel in the image
  Show the classification result to the user; ask them to paint some labels again to improve classification
  Loop until the user is satisfied with the classification
  return the whole pixel classifier, features + random forest, to be applied directly to other images
```

3 Other applications of deep learning in medical imaging [Litjens et al., 2017

3.1 Classification of images from medical exams

3.2 Content-based image retrieval

Browsing massive databases, for instance retrieving similar case histories. Feature extraction is very useful here.

3.3 Generating text reports from images [Schlegl et al., 2015]

References

[Berg et al., 2019] Berg, S., Kutra, D., Kroeger, T., Straehle, C. N., Kausler, B. X., Haubold, C., Schiegg, M., Ales, J., Beier, T., Rudy, M., Eren, K., Cervantes, J. I., Xu, B., Beuttenmueller, F., Wolny, A., Zhang, C., Koethe, U., Hamprecht, F. A., and Kreshuk, A. (2019). ilastik: interactive machine learning for (bio)image analysis. *Nature Methods*.

[Besl and McKay, 1992] Besl, P. J. and McKay, N. D. (1992). A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(2):239–256.

[Cao et al., 2014] Cao, T., Zach, C., Modla, S., Powell, D., Czymmek, K., and Niethammer, M. (2014). Multi-modal registration for correlative microscopy using image analogies. *Medical Image Analysis*, 18(6):914–926.

[Chen and Medioni, 1991] Chen, Y. and Medioni, G. G. (1991). Object modeling by registration of multiple range images. In *Proceedings of the 1991 IEEE International Conference on Robotics and Automation, Sacramento, CA, USA, 9-11 April 1991*, pages 2724–2729.

[Cheng et al., 2018] Cheng, X., Zhang, L., and Zheng, Y. (2018). Deep similarity learning for multimodal medical images. *CMBBE: Imaging & Visualization*, 6(3):248–252.

[Cour et al., 2007] Cour, T., Srinivasan, P., and Shi, J. (2007). Balanced graph matching. In *Advances in Neural Information Processing Systems*, pages 313–320.

[Elad and Aharon, 2006] Elad, M. and Aharon, M. (2006). Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. Image Processing*, 15(12):3736–3745.

[Fischler and Bolles, 1981] Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395.

[Frank and Wolfe, 1956] Frank, M. and Wolfe, P. (1956). An algorithm for quadratic programming. *Naval research logistics quarterly*, 3(1-2):95–110.

[Gertz and Wright, 2003] Gertz, E. M. and Wright, S. J. (2003). Object-oriented software for quadratic programming. *ACM Trans. Math. Softw.*, 29(1):58–81.

[Hertzmann et al., 2001] Hertzmann, A., Jacobs, C. E., Oliver, N., Curless, B., and Salesin, D. (2001). Image analogies. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2001, Los Angeles, California, USA, August 12-17, 2001*, pages 327–340.

[Huang et al., 2016] Huang, X., Zhang, J., Fan, L., Wu, Q., and Yuan, C. (2016). A systematic approach for cross-source point cloud registration by preserving macro and micro structures. *IEEE Transactions on Image Processing*, PP.

[Johnson et al., 2015] Johnson, H. J., McCormick, M. M., and Ibanez, L. (2015). *The ITK Software Guide Book 2: Design and Functionality-Volume 2*. Kitware, Inc.

[Kan, 2017] Kan, A. (2017). Machine learning applications in cell image analysis. *Immunology and cell biology*, 95(6):525–530.

[Klein et al., 2010] Klein, S., Staring, M., Murphy, K., Viergever, M. A., and Pluim, J. P. W. (2010). elastix: A toolbox for intensity-based medical image registration. *IEEE Trans. Med. Imaging*, 29(1):196–205.

[Li and Liu, 2009] Li, H. and Liu, F. (2009). Image denoising via sparse and redundant representations over learned dictionaries in wavelet domain. In *Proceedings of the Fifth International Conference on Image and Graphics, ICIG 2009, Xi'an, Shanxi, China, 20-23 September 2009*, pages 754–758.

[Litjens et al., 2017] Litjens, G. J. S., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., van der Laak, J. A. W. M., van Ginneken, B., and Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42:60–88.

[Marks, 2018] Marks, C. (2018). Identifying nuclei in divergent images: Data science bowl 2018.

[Papazov and Burschka, 2011] Papazov, C. and Burschka, D. (2011). Stochastic global optimization for robust point set registration. *Computer Vision and Image Understanding*, 115(12):1598–1609.

[Papon et al., 2013] Papon, J., Abramov, A., Schoeler, M., and Wörgötter, F. (2013). Voxel cloud connectivity segmentation - supervoxels for point clouds. In *2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013*, pages 2027–2034.

[Rusu and Cousins, 2011] Rusu, R. B. and Cousins, S. (2011). 3d is here: Point cloud library (PCL). In *IEEE International Conference on Robotics and Automation, ICRA 2011, Shanghai, China, 9-13 May 2011*.

[Schlegl et al., 2015] Schlegl, T., Waldstein, S. M., Vogl, W., Schmidt-Erfurth, U., and Langs, G. (2015). Predicting semantic descriptions from medical images with convolutional neural networks. In *Information Processing in Medical Imaging - 24th International Conference, IPMI 2015, Sabhal Mor Ostaig, Isle of Skye, UK, June 28 - July 3, 2015, Proceedings*, pages 437–448.

[Simonovsky et al., 2016] Simonovsky, M., Gutiérrez-Becker, B., Mateus, D., Navab, N., and Komodakis, N. (2016). A deep metric for multimodal registration. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2016 - 19th International Conference, Athens, Greece, October 17-21, 2016, Proceedings, Part III*, pages 10–18.

[Toledo Acosta et al., 2018] Toledo Acosta, B. M., Heiligenstein, X., Malandain, G., and Bouthemy, P. (2018). Intensity-based matching and registration for 3d correlative microscopy with large discrepancies. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pages 493–496. IEEE.

[Tu and Bai, 2010] Tu, Z. and Bai, X. (2010). Auto-context and its application to high-level vision tasks and 3d brain image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(10):1744–1757.

[Wohlkinger and Vincze, 2011] Wohlkinger, W. and Vincze, M. (2011). Ensemble of shape functions for 3d object classification. In *2011 IEEE International Conference on Robotics and Biomimetics, ROBIO 2011, Karon Beach, Thailand, December 7-11, 2011*, pages 2987–2992. IEEE.

[Wu et al., 2013] Wu, G., Kim, M., Wang, Q., Gao, Y., Liao, S., and Shen, D. (2013). Unsupervised deep feature learning for deformable registration of MR brain images. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2013 - 16th International Conference, Nagoya, Japan, September 22-26, 2013, Proceedings, Part II*, pages 649–656.

[Zaslavskiy et al., 2008] Zaslavskiy, M., Bach, F., and Vert, J.-P. (2008). A path following algorithm for the graph matching problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2227–2242.

[Zhou and De la Torre, 2013] Zhou, F. and De la Torre, F. (2013). Deformable graph matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2922–2929.