University of Liège
Faculty of Engineering

Master Thesis

A workflow for large-scale computer-aided cytology and its applications.

Author : Romain Mormont

Supervisor : Prof. Pierre Geurts

Master thesis submitted for the degree of

MSc in Computer Science and Engineering

Academic year 2015-2016

Contents

Summary

Acknowledgement

Chapter 1

Introduction

Chapter 2

Object detection in large images

2.1 General problem

2.1.1 Formulation

Generic formulation of the object detection problem

2.1.2 Implementation issues

What issues an implementor could face when trying to implement object detection in large images

2.1.3 Related works

What solutions are usually presented in the litterature to solve those problems (shallow overview as this is a wide topic)

2.2 Cytology

2.2.1 An object detection problem ?

Why it is an instance of the ”object detection in large images” problem

2.2.2 The thyroid case

Explanation of the Thyroid case

Chapter 3

A generic workflow : Segment Locate Dispatch Classify

3.1 Principle

3.1.1 History and first developments

The Segment-Locate-Dispatch-Classify (SLDC) workflow was first imagined by ?? Jean-Michel Begon ?? as a generalization of the work on thyroid nodule malignancy detection made in [Deb13]. In the context of this master thesis, the author had implemented a processing workflow for detecting cells with inclusion and proliferative patterns (see ?? (thyroid)) in digitized thyroid punctions slides. The cells and patterns were detected by segmenting the images and then classified using machine learning techniques. As explained in the Section ?? (thyroid), patterns could themselves contain cells with inclusion. Therefore, the author implemented a second processing workflow to detect those cells. This workflow was similar to the first because it relied on a segmentation algorithm to isolate cells in patterns and then used machine learning to assess their malignity.

From those workflows, a common pattern emerged: performing detection using a segmentation algorithm and then classifying the detected objects using machine learning techniques. This pattern was at the root of the SLDC workflow idea.

3.1.2 Algorithm

The workflow is a meta-algorithm[1] that detects and classifies objects contained in images. Particularly, given an image $\mathcal{I}$ as input, it is expected to output the information about the objects of interest in this image. Those information include the shape of the object, its location in the image as well as a classification label. Formally, the workflow can be seen as an operator $\mathcal{W}$:

Definition 1. *Let $\mathcal{W}$ be an operator such that*

$$\mathcal{W}(\cdot) : \mathcal{I} \rightarrow \{(o_1, C_1), ..., (o_N, C_N)\} \tag{3.1}$$

where N is the number of objects of interest in $\mathcal{I}$ and (o_i, C_i) is a tuple. The first element of this tuple, o_i, is a representation of the information (shape and location) about the i^{th} object of interest found in $\mathcal{I}$ and the second, C_i, its classification label.

[1] In this context, a meta-algorithm is an algorithm that coordinates the execution of other algorithms.

It is worth noting that genericity is of the essence. That is, the meta-algorithm should be able to solve the widest possible range of object detection and classification problems. Moreover, as explained in Section 3.1.1, it should produce those outputs using image segmentation and machine learning. As far as the segmentation is concerned, genericity is usually hard to obtain because of the high variability of images across different problems. In order to ensure genericity, the workflow doesn't impose a particular segmentation procedure but expects the implementor to provide one that suits the problem. The same goes for the classification models used for predicting the labels of the objects.

In the subsequent sections, some additional operators are defined and used to build the $\mathcal{W}$ operator. First, a basic version of the algorithm is presented and then refined in order to achieve an acceptable level of genericity.

3.1.3 Additional operators

Segmentation is the first operation applied to the image. This step of the algorithm is where the detection is actually carried out:

Definition 2. *Let $\mathcal{S}$ be the* ***segment*** *operator. It is applied to an image $\mathcal{I}$ and produces a binary mask $\mathcal{B}$. The pixel b_{ij} of $\mathcal{B}$ is 1 if the pixel p_{ij} of $\mathcal{I}$ is located in an object of interest, otherwise it is 0. Formally:*

$$\mathcal{S}(\cdot) : \mathcal{I} \rightarrow \mathcal{B} \tag{3.2}$$

While the segmented image theoretically contains the necessary information about the detected objects (i.e. shape and position in the image), the format of this information is inconvenient to query mostly because it is embedded into the binary mask and a single object cannot be trivially extracted. An intermediate step that would convert this information into a more convenient format is therefore needed. This format should encode both the shape of the object and its position in the image. It appears that polygons match this specification.

Definition 3. *Let $\mathcal{L}$ be the* ***location*** *operator. It is applied to a binary mask and produces a set of polygons encoding the shapes and positions of every object in the image. Formally:*

$$\mathcal{L}(\cdot) : \mathcal{B} \rightarrow \{P_1, ..., P_N\} \tag{3.3}$$

where $\mathcal{B}$ is a binary mask as defined in Definition 2, N is the number of objects of interest in $\mathcal{B}$ and P_i is the polygon representing the geometric contour of the i^{th} object in $\mathcal{B}$.

The final step of the workflow is the object classification and is performed by a classifier. This object is passed a representation of the object (e.g. image, geometric information,...) and produces a classification label. In this theory, there is no restriction about the nature or representation of the objects processed by the classifiers.

Definition 4. *Let $\mathcal{T}$ be the* ***classifier*** *operator. It is applied to an object of interest and produces a classification label. Formally:*

$$\mathcal{T}(\cdot) : o \rightarrow C \tag{3.4}$$

where o is the object and C, the classification label.

Definition 5. *Let $\mathcal{T}^*$ be an extension of $\mathcal{T}$ which is given a set of objects and produces labels for all of them. Formally:*

$$\mathcal{T}^*(\cdot) : \{o_1, ..., o_N\} \rightarrow \{C_1, ..., C_N\} \tag{3.5}$$

3.1.4 Single segmentation, single classifier

The most simple construction of $\mathcal{W}$ would be the composition of the operators defined in Section 3.1.3. Particularly, the composition $\mathcal{S} \circ \mathcal{L}$ would produce the polygons representing the objects and the composition $\mathcal{S} \circ \mathcal{L} \circ \mathcal{T}^*$ would produce the labels. Formally:

$$\mathcal{S} \circ \mathcal{L} : \mathcal{I} \to \{P_1, ..., P_N\} \tag{3.6}$$

$$\mathcal{S} \circ \mathcal{L} \circ \mathcal{T}^* : \mathcal{I} \to \{C_1, ..., C_N\} \tag{3.7}$$

As explained in Section 3.1.2, the definition of $\mathcal{S}$ and $\mathcal{T}^*$ would be left at the implementor's hands. As far as the $\mathcal{L}$ operator is concerned, it could be imposed by the workflow without loss of genericity. Such an construction of $\mathcal{W}$ could already solve any object detection and classification problem on image in which the labels can be predicted by a single classifier. However, in some cases, one classifier is not enough. This happen, for instance, when the image contains objects of very different nature and using several classifiers would yield better results than using a single one. An extension is therefore needed.

3.1.5 Single segmentation, several classifiers

In this attempt to construct a generic $\mathcal{W}$ operator, the image is assumed to contain M distinct types of objects and the workflow uses M classifiers (the i^{th} classifier being noted as $\mathcal{T}_i$ with $i \in \{1, ..., M\}$) to classify those objects. As an object should only be processed by one classifier, the workflow has to be added a new step which consists in dispatching each polygon to its most appropriate classifier.

Definition 6. *Let $\mathcal{D}$ be the dispatch operator. It is applied to a polygon and produces an integer which identifies the most appropriate classifier for processing this polygon:*

$$\mathcal{D}(\cdot) : P \to i, i \in \{1, ..., M\} \tag{3.8}$$

This step being problem dependent, it is the responsibility of the implementor to define the rules used for dispatching the polygons. However, the format of these rules can be defined.

Definition 7. *Let $p_1, ..., p_M$ be a set of M predicates which associate truth values to polygons:*

$$p_i(\cdot) : P \to t \in \{true, false\}, i \in \{1, ..., M\} \tag{3.9}$$

where p_i is the predicate associated with the i^{th} classifier. The polygon P is dispatched to a classifier $\mathcal{T}_i$ if p_i associates true to this polygon. To avoid dispatching an object to several classifier, the predicates should verify the following property:

$$p_i = true \Leftrightarrow p_j = false, \forall j \neq i \tag{3.10}$$

Given this format, the $\mathcal{D}$ operator can be trivially constructed as it returns i if p_i is *true*. The algorithm resulting from this construction of $\mathcal{W}$ starts the same way as in Section 3.1.4: the image is applied the segment and locate operators (see Equation 3.6). Then, the resulting polygons are dispatched and classified to produce the classification label. The resulting algorithm is summarized in Algorithm 1.

While the range of problems that can be solved using this algorithm has been increased compared to the version with a single classifier, there are still some problems that cannot be. In particular, if some objects are included into other bigger objects, they won't be considered as independent objects.

Algorithm 1. *Construction of the $\mathcal{W}$ operator with a single segmentation and several classifiers.*

1. *Apply the $\mathcal{S} \circ \mathcal{L}$ composition to the input image $\mathcal{I}$ to extract the objects of interest as the set of polygons $S_p \leftarrow \{P_1, ..., P_N\}$*
2. *Initialize the labels set $L \leftarrow \emptyset$*
3. *For each polygon $P \in S_p$:*
   (a) *Compute the classification label $C \leftarrow \mathcal{T}_{\mathcal{D}(P)}(P)$*
   (b) *Place the label in the labels set $L \leftarrow L \cup \{C\}$*
4. *Build and return objects and labels set $S_p \times L$ ($\times$ being a cartesian product).*

3.1.6 Chaining workflows

To take into account the case when objects of interest can be included into objects of bigger size in the image, the construction of $\mathcal{W}$ has to be improved. A way of solving this issue is to chain executions of Algorithm 1. While the first execution would extract objects of interests from the full image $\mathcal{I}$, the subsequent steps would have the responsibility to extract the included objects.

3.2 Implementation

3.2.1 Framework

Describe what we expect from a framework implementing the workflow (parallism, easy to use,...)

As explained in Section 3.1.1, a first version of the workflow was developed in the context of Cytomine.

3.2.2 Technologies

3.2.3 Software architecture

Detail of the software architecture

3.2.4 How to use the framework

A toy example: finding disks in an image with grey background and guessing whether they're black or white

Chapter 4

SLDC at work : the thyroid case

4.1 Cytomine

Presentation of cytomine

4.2 Implementation issues

Presentation of implementation issues related to the thyroid case (image size, over HTTP, image quality, human annotation vs computer annotation, presence of inclusions in patterns, dispatching ...)

4.3 Implementation

Actual implementation of the processing using the workflow

4.4 Performance analysis

4.4.1 Detection

4.4.2 Execution time

Chapter 5

Conclusion

Notations

Image :

$\mathcal{I}$	An image
w	The width of an image
h	The height of an image
c	The number of channels of an image
$\mathcal{I}_{hw}$	An two dimensional image of width w and height h
p_{ij}	A pixel at row i and column j of a two dimensional image
$\mathcal{B}$	A binary image
$b_{ij} \in \{0, 1\}$	A pixel of a binary image
P	A polygon

Machine learning :

$T(\cdot)$	A classifier
C	A classification label

List of Tables

List of Figures

Bibliography

[Deb13] Antoine Deblire. “Segmentation et classification automatiques de cytoponctions de la thyroïde.” fr. MA thesis. Université de Liège, Liège, Belgique, 2013, p. 74.