An Agent-based System for Printed/Handwritten Text Discrimination

Florence Cloppet[1], Pavlos Moraitis[1], and Nicole Vincent[1]

LIPADE, Paris Descartes University, France
florence.cloppet@mi.parisdescartes.fr,
pavlos@mi.parisdescartes.fr,
nicole.vincent@mi.parisdescartes.fr

Abstract. The handwritten/printed text discrimination problem is a decision problem usually solved after a binarization of grey level or color images. The decision is usually made at the connected component level of a filtered image. These image components are labeled as *printed* or *handwritten*. Each component is represented as a point in a n dimensional space based on the use of n different features. In this paper we present the transformation of a (state of the art) traditional system dealing with the handwritten/printed text discrimination problem to an agent-based system. In this system we associate two different agents with the two different points of view (i.e. linearity and regularity) considered in the baseline system for discriminating a text, based on four (two for each agent) different features. We are also using argumentation for modeling the decision making mechanisms of the agents. We then present experimental results that compare the two systems by using images of the IAM handwriting database. These results empirically prove the significant improvement we can have by using the agent-based system.

1 Introduction

Automated image analysis is an important topic in artificial intelligence. The *handwritten/printed* text discrimination problem is a specific problem in the field of image analysis. Several systems in the literature (see e.g. [12], [9],[13]) have proposed different solutions for this particular problem. The main idea is to consider only small elements of the document image such as textual parts that form a *connected component (CC)*. Such elements are the characters in printed texts and the words in cursive handwritten texts. The connected components are labeled using three labels namely "printed", "handwritten" or "other", depending on the type of the document being processed (e.g. document with or without images). Usually CCs are extracted from a binarized image. The labeling process may be applied at different levels according to the statistical approach used. Thus some systems are considering a block of text, a paragraph, a whole page or few lines to make the measurements statistically consistent. The characterization of these components is based on several features (e.g. size, density, Gabor filters, Run-length, SIFT, bag of visual words, etc.) and therefore a high dimensional space is needed for their representation. Then, a two class classifier is learnt. Sometimes, a post processing phase is considered using a more global view. In that case the labeled components' positioning in the space may be modeled by using for example a Markov random field.

In [7] we have presented one of those systems. The specificity of the developed approach relies on a small number of features considered as meaningful and a quantization of their evolution. The features are related to the description of a written text style (i.e. aspect of the trace of the writing stroke on the sheet of paper) and can be divided in two classes namely features linked to the (more or less) *linear aspect* of the strokes and features linked to the (more or less) *regular aspect* of the components. Both points of view give hints on the nature of a text.

In this paper, we first present the *handwritten/printed text* discrimination problem and we analyze the system proposed in [7]. We then motivate our decision to use agent technology for solving the above problem and we prove the added value of our approach by presenting an agent-based approach of the handwritten/printed text discrimination problem as formulated in [7]. More particularly we show that this problem can be modeled as a *distributed decision making problem* by presenting a detailed description of the agents' architecture, the way the agents reason for making individual decisions by using argumentation and finally the way they interact through a bilateral dialogue for solving collectively the given problem. Finally we present experimental results that empirically prove that our agent-based approach significantly improves the performances of the traditional system proposed in [7]. For this reason we have compared both systems based on 2138 connected components extracted from 25 randomly chosen images from the IAM handwriting database[1].

2 Basics

2.1 Argumentation

Argumentation (see e.g. [2]) can be abstractly defined as the formal interaction of different conflicting arguments for and against some conclusion due to different reasons and provides the appropriate semantics (see e.g. [6]) for resolving such conflicts and determining which are the winning arguments. Thus, it is very well suited for implementing decision making mechanisms. Moreover when the decisions are involving dynamic preferences we need a specific type of argumentation frameworks. For this reason we have chosen the framework proposed in [8]. This framework has been applied in a successful way in different applications (see e.g. [11]) involving dynamic preferences and it is supported by an open source software called Gorgias[2].

In this framework the argumentation theories are represented at three levels. The *object level arguments* representing the decisions (or the actions) an agent

[1] http://www.iam.unibe.ch/fki/databases/iam-handwriting-database

[2] http://www.cs.ucy.ac.cy/ nkd/gorgias/

can undertake in a specific domain of application and *priority arguments* expressing preferences on the object level arguments in order to resolve possible conflicts. *Higher order priority arguments* are also used in order to resolve potential conflicts between priority arguments of the previous level. This framework allows for the representation of dynamic preferences under the form of dynamic priorities over arguments and uses Dung semantics [6].

An argumentation theory is a pair $(\mathcal{T}, \mathcal{P})$ whose sentences are formulas in the background monotonic logic $(\mathcal{L}, \vdash)$ of the form $L \leftarrow L_1, \ldots, L_n$, where $L, L_1, \ldots, L_n$ are positive or negative ground literals. Rules in $\mathcal{T}$ represent the object level arguments. Rules in $\mathcal{P}$ represent priority arguments where the head L refers to an (irreflexive) *higher priority* relation. L has the general form $L = h\_p(rule1, rule2)$ where h_p stands for higher priority. The derivability relation, $\vdash$, of the background logic is given by the simple inference rule of modus ponens. Thus, more formally we have:

Definition 1. *[8] An agent's* **argumentative policy theory or theory,** T**,** *is a triple* $T = (\mathcal{T}, \mathcal{P}_R, \mathcal{P}_C)$ *where the rules in* $\mathcal{T}$ *do not refer to* h_p*, all the rules in* $\mathcal{P}_R$ *are priority rules with head* $h\_p(r_1, r_2)$ *s.t.* $r_1, r_2 \in \mathcal{T}$ *and all rules in* $\mathcal{P}_C$ *are priority rules with head* $h\_p(R_1, R_2)$ *s.t.* $R_1, R_2 \in \mathcal{P}_R \cup \mathcal{P}_C$*.*

2.2 Image Analysis

In this section we discuss basic concepts related to the image analysis problem. An image is composed of a set of pixels each of them having a different luminance level that corresponds to a particular color (i.e. from 0 for black color to 255 for white). Their spatial distribution is represented in a matrix. An image is then represented as a rectangular array where the indexes refer to the spatial location of a pixel while the elements' values refer to the color of the pixels. A document image is a specific type of image, representing a paper document through an array structure. A binary image is characterized by the use of only two luminance levels. This limitation is quite convenient when studying document images where the text appears in black and the background in white. Then a binary image corresponds to a set of objects that are the connected components of the black pixels. In printed texts, connected components are basically the characters. Nevertheless, an "i" letter comprises two connected components as the dot is a connected component by itself.

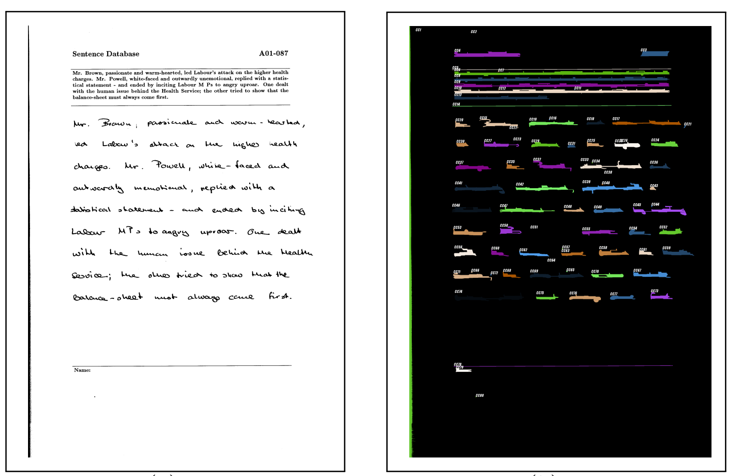
3 The Handwritten/Printed Text Discrimination Problem

A document is a juxtaposition of different media (i.e. text or image for instance) that have different appearances. Thus automatization of a document image processing greatly depends on the content of these media. In the field of document image analysis, the different parts of a document are not processed in the same way. For example illustrative figures are not processed as are texts or tables [10].

Herein we are interested in the discrimination between printed and handwritten texts. Although they are both texts, they refer to different sources of knowledge. To overcome the semantic gap between a word image and its meaning, an optical character recognition software (OCR) is used. However, we cannot use the same OCR software for both types of texts (i.e. printed and handwritten). Current electronic document management systems (EDMS) don't apply for hybrid documents (e.g. printed documents with handwritten annotations). Thus, adding an automated printed/handwritten discrimination step in the management of electronic document would allow processing such hybrid documents.

3.1 The Baseline System

The handwritten/printed text discrimination is a problem involving two possible decisions (or two-class problem) usually solved after a binarization of grey level images. The decision is usually made at the *connected component* (CC) level of a binarised image (see Figure 1)and they are labeled as *printed* or *handwritten*. n features are computed enabling the representation of each CC as a point in a n dimensional space. These features refer to the texture or the shape of the CCs and are computed through image transformations (e.g. wavelets, Haralick features). The vectors of features' values are the input of a classifier. Then classifiers such as SVM (see e.g. [9]), k nearest neighbors (see e.g. [5]), or naive Bayes classifier (see e.g. [5]) can be built, during a training step.



(a) (b)

Fig. 1. *Working connected components* of the image iam087: (a) Binarized Document Image (image 087 of the IAM database) (b) Its *Connected Components (WCCs)* .

As previously said, in our work we use as baseline system the framework we proposed in [7]. This framework uses a small number of features that may refer to two different points of view, namely *regularity* and *linearity*. The labeling process is applied on connected components (CCs) extracted from the binarized image on which a Run Length Smoothing Algorithm ($RLSA$) transformation has been applied in order to have *Working* CCs (or $WCCs$) with a significant area (see Figure 1). Moreover in this work we have shown that although the number of features we use is very small, the obtained results are competitive with the results of systems using several hundreds of features. For helping the reader we will explain here these two points of view.

Linearity is computed on connected components using two parameters:

1. a parameter denoted a representing the error level of the polygonal approximation of a connected component's (CC) contour (see Figure 2). This parameter takes two values i.e. $a \in \{1, 5\}$;
2. a parameter denoted c comparing the value of a feature related to a component with the value of the same feature related to the union of the component itself (corresponding to value i) and its symmetrical one wrt a horizontal straight line (corresponding to value s). This parameter takes therefore also two values i.e. $c \in \{i, s\}$ (see Figure 3).

From a polygonal approximation of a component's contour, a set of straight line segments is extracted and then a histogram H of the segments' lengths is considered. Two features are then computed, namely the maximum of the segment lengths $SM_{a,c}(H)$ and an estimation $L_{a,i}(H)$ of the histogram shape as defined in formula (1). The decision process involves the comparison of the two values $L_{5,i}(H)$ and $L_{1,i}(H)$ (used in formula (2)). It also involves the comparison of $SM_{5,i}(H)$ and $SM_{5,s}(H)$ (used in formula (3)).

Fig. 2. Two different polygonal approximations (a=1 on the left, a=5 on the right) from [7]

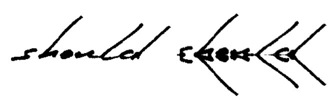


Fig. 3. The zone of interest (i-left) and its union with its symmetrical part (s-right)

$$\mathrm{L}_{a,i}(H) = \frac{\displaystyle\sum_{l=1}^{SM_{a,i}(H)} l \cdot (H(l) - H(l-1)) \cdot \chi_{[0,+\infty[}(H(l) - H(l-1))}{\displaystyle\sum_{l=0}^{SM_{a,i}(H)} H(l)} \tag{1}$$

The point of view of *regularity* is based on two features as well. These features are computed from the upper and lower profile of a component and are called *upper regularity* R_U and *lower regularity* R_L. Each profile is a sequence of points characterized by their vertical position. R_U and R_L are defined as the standard deviation of the points' vertical positions in the upper and lower profile respectively (used in formulas (4) and (5)). In the current work we consider the same definition of features as in [7]. More precisely the evolution of the L value relies on the values of parameter a which correspond to the values of the *precision* parameter in the Wall algorithm [15]. We also consider the evolution of SM when using i and s (see above).

$$\mathrm{L}_{5,\mathrm{i}}(H) \;>\; \mathrm{LT}_1 \cdot \mathrm{L}_{1,\mathrm{i}}(H) \tag{2}$$

$$\mathrm{SM}_{5,\mathrm{i}}(\mathrm{H}) \;>\; \mathrm{LT}_2 \cdot \mathrm{SM}_{5,\mathrm{s}}(\mathrm{H}) \tag{3}$$

$$\mathrm{R}_{\mathrm{U}} \;>\; \mathrm{RT}_1 \tag{4}$$

$$\mathrm{R}_{\mathrm{L}} > \mathrm{RT}_2 \tag{5}$$

The decision making is achieved after a learning step providing four threshold values i.e. $LT_1 = 3$, $LT_2 = 1.5$, $RT_1 = 0.02$, $RT_2 = 0.0045$. The decision function based on those values is a piecewise linear function. Once the four previous boolean values have been computed (see above formulas (2), (3), (4), (5)), the decision is taken according to the following rule (see formula (6)).

$$\textit{A component is labeled as handwritten if } \; (2)\,\mathrm{AND}\,(3)\,\mathrm{AND}\,(\,(4)\,\mathrm{OR}\,(5)\,) \tag{6}$$

3.2 The Baseline System vs State of the Art

The comparison between different methods is difficult as the used databases are not public and therefore they may differ from one study to another. The evaluation can be done at word, pseudo-word or pixel level. Furthermore the results depend on the sets used for the learning and validation steps. The system we presented in [7] was developed in an industrial environment and it has been evaluated on a large dataset of real documents used by the company. The system was run by the company and evaluated at pixel level. It has been also compared to another system [1] chosen by the same company and the results are presented in Table 1.

Table 1. Comparison of systems proposed in [1] and [7] (presented in [7])

	Baseline system [7]	Belaid et al. system [1]
Text entity	Pseudo-word	Pseudo-word
Descriptors	4	137
Classifiers	Decision rules	SVM
Regularization	kNN	kNN
Database	Industrial dataset	Industrial dataset
Recognition rate (%)	90.15	89.05

As we can see in the above table, the baseline system slightly outperforms the system proposed in [1] although it uses far fewer features than this system. That means that the features considered in [7] are very meaningful.

4 Why Using Agent Technology?

4.1 Motivation

Our motivation for using agent technology is based on four drawbacks we have observed regarding the baseline system we proposed in [7] (and discussed in the previous section) but also other traditional systems (see e.g. [9], [1], [12],[13]) using several points of view:

1. The different points of view (e.g. regularity and linearity in our case) are not independently represented.
2. The parameters of the decision functions associated with these points of view cannot change after the learning phase.
3. There is a need to make default decisions in a majority of situations but also to adapt these decisions to particular contexts
4. There is a lack of explanation of the decisions taken by the system.

In this paper we propose a novel approach which takes into consideration the four aforementioned drawbacks. More precisely, by associating an agent with a point of view we are able to detect situations where the decisions do not coincide (i.e. the agents have an initial disagreement). This is an important information as it can trigger in both agents a fine tuning of their initial decision models. In that case the discrimination problem is transformed into a *distributed decision making* problem where agents are looking for an agreement (i.e. by proposing the same decision namely "printed" or "handwritten") through a bilateral dialogue. During this dialogue their initial decision models are evolving when a (transitory) disagreement occurs at the end of a round. Finally, the last two drawbacks are taken into consideration by using argumentation for modeling the decision making mechanisms of the agents. The framework we are using [8] allows for the modeling of default but also context dependent decisions when this is needed. Argumentation also allows for decisions explanation as the framework we are using can present a trace of the reasoning that agents have followed for making a decision.

4.2 Our Approach

For illustrating how our system works we will consider the decision making problem considered in [7] where two different decisions (or opinions), referring to two different points of view, have to be reconciliated. We may consider that each decision is based on different numerical features (e.g. presence of straight line segments on the contour of the writing, upper and lower regularity profiles of the writing, etc.) extracted from an image. Each decision is made by using a decision function. In our case the representation space for both points of view is a two dimensional space and thus the decision function can be represented by a curve in a two dimensional space. This function is built by using a learning technique. The image under analysis is represented by a point in this space. A decision is therefore depending on the position of this point wrt to the curve (see Figure 4). The initial situation is represented by a solid line (see Figure 4). We have two possible decisions, decision D_1 (i.e. the image represents a printed text) and decision D_2 (i.e. the image represents a handwritten text).

Let's now describe our approach where we associate an agent with a point of view. We consider two agents α and β associated with the initial decisions D_1 and D_2 respectively. As these decisions are different, the agents will enter into a dialogue. In [7] these decisions cannot change during the process but in our approach they can. More particularly, the agents cannot challenge the structure of the decision function but they can move the curve in one direction so that the point representing the image gets closer to the curve. As the curve's shifting cannot be done in a continuous way, we consider a *unit measure* representing the "distance" between two successive curve's positions. Every shifting of a curve corresponds to an action undertaken by each agent when a disagreement is detected after the exchange of their individual decisions. This action refers to the change of some parameters of the decision function within the chosen family of functions used. As the iso decision curves are not regularly positioned in the space, we have decided to model the shifting of the decision curves by using two different parameters. Each exchange of the individual decisions is considered as a dialogue round. This process continues until a consensus is reached. An agent changes his opinion/decision when the image representation is found on the other side of the moving curve.

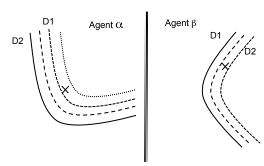


Fig. 4. Initial position (solid line) and shifting (dotted lines) of the decision function for α and β during the dialogue

As we can see in Figure 4, agent α needs three steps for changing his decision (illustrated by the dotted line curves) while agent β needs only two. In the current example we can observe that agent β has changed his decision after two dialogue rounds. The final decision will therefore be decision D_1 which is the initial decision of agent α. We therefore associate more confidence to the agent who has more resisted to a change of his decision.

However, we can have situations where both agents might change their decisions in the same dialogue round. In that case our approach proposes the adaptation of the unit measure in a recursive way. This case is illustrated in the upper part of the Figure 5 where both agents change their decisions after the second round of the dialogue. In this situation agents have to get back to the

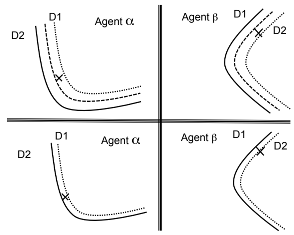


Fig. 5. Change of the unit measure value when agents change decision simultaneously in a dialogue round

previous state of the dialogue and to decrease the unit measure before starting a new dialogue round as illustrated in the bottom of Figure 5. The dialogue terminates either with an agreement (i.e. when one of the agents changes his decision and agrees with the other) or with a final disagreement (i.e. when the unit measure attains a minimum value without an agreement to be found). In that case, the decision is taken randomly. The overall architecture of our agent-based approach can be represented as in Figure 6.

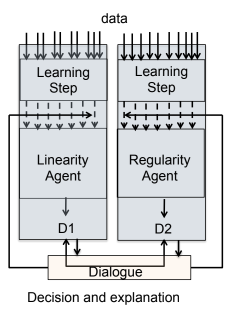


Fig. 6. Representative architecture of our proposal

5 The Agent-based System

In this section we will present the agent-based system we have designed for implementing the approach we have proposed in 4.2.

5.1 Decision Theories of Agents

In this section we present the transformation of the baseline system presented in [7] into a multi-agent system involving two autonomous agents associated with the two different points of view presented in section 3.1, namely *linearity* and *regularity*. We will call the agents, *Linearity* agent and *Regularity* agent, respectively. For representing the knowledge of these agents we have used the formulas (2), (3), (4), (5) and the decision rule (6). As presented in 3.1, the first two conditions (i.e. 2 and 3) are referring to the linearity point of view, while the last two conditions (i.e. 4 and 5) to the regularity point of view. Thus the decisions (i.e. "printed" or "handwritten") taken in the baseline system by using the decision rule 6, will, in our approach, be taken through a bilateral dialogue between Linearity Agent (LA) and Regularity Agent(RA). The final decision will thus be corresponding to an agreement between the two agents reached after one or several rounds of dialogue.

For representing the decision theories of the two agents we translated the corresponding formulas into argumentation theories by using the argumentation framework proposed in [8] (and discussed in section 2.1). The use of argumentation allows to represent in a more explicit way the different scenarios that are generated when the features $L_{5,i}(H)$, $L_{1,i}(H)$, $SM_{5,i}(H)$ and $SM_{5,s}(H)$ are taking specific values wrt the thresholds LT_1 and LT_2 for Linearity Agent. The same holds for the scenarios that are generated when the features R_U and R_L are taking specific values wrt the thresholds RT_1 and RT_2 for Regularity Agent. This representation puts explicitly in evidence the possible dilemmas of the agents by detecting conflicting situations (i.e. when both decisions namely "handwritten" and "printed" are possible) and allows to use default (or generic) and contextual knowledge for solving these conflicts.

The *Linearity* agent theory is as follows. The two possible decisions are $d_1^{Lin} = "Printed"$ and $d_2^{Lin} = "Handwritten"$.

$r_1 : d_1^{Lin} \leftarrow (SM_{5,s}(H)/SM_{5,i}(H)) > LT_1$
$r_2 : d_2^{Lin} \leftarrow (L_{5,i}(H)/L_{1,i}(H)) > LT_2$
$r_3 : d_2^{Lin} \leftarrow (SM_{5,s}(H)/SM_{5,i}(H)) \leq LT_1$
$r_4 : d_1^{Lin} \leftarrow (L_{5,i}(H)/L_{1,i}(H)) \leq LT_2$
$R_1 : h\_p(r_1, r_2) \leftarrow true$
$R_2 : h\_p(r_4, r_3) \leftarrow true$

The *Regularity* agent theory is as follows. The two possible decisions are $d_1^{Reg} = "Printed"$ and $d_2^{Reg} = "Handwritten"$.

$r_1 : d_1^{Reg} \leftarrow R_U \leq RT_1$
$r_2 : d_1^{Reg} \leftarrow R_L \leq RT_2$
$r_3 : d_2^{Reg} \leftarrow R_L > RT_2$
$r_4 : d_2^{Reg} \leftarrow R_U > RT_1$
$R_1 : h\_p(r_3, r_1) \leftarrow true$
$R_2 : h\_p(r_4, r_2) \leftarrow true$
$R_3 : h\_p(r_1, r_3) \leftarrow H_{CC} \leq HT, W_{CC} > 20 * HT$
$R_4 : h\_p(r_2, r_4) \leftarrow H_{CC} \leq HT, W_{CC} > 20 * HT$
$C_1 : h\_p(R_3, R_1) \leftarrow true$
$C_2 : h\_p(R_4, R_2) \leftarrow true$

The above theories show how we can represent the conflicting knowledge associated with each point of view (i.e. linearity and regularity) and how we can put in evidence the contradictory decisions that can be taken when some situations may arrive simultaneously. These conflicting situations are described by rules r_1 and r_2 and rules r_3 and r_4 in the linearity theory and also by rules r_1 and r_3 and rules r_2 and r_4 in the regularity theory. However, we note that the experts are able to prioritize these conflicting situations and to solve the generated conflicts. This is done with rules R_1 and R_2 in the linearity theory where the priority is given to the decision "printed" and the rules R_1 and R_2 in the regularity theory where the priority is given to the decision "handwritten". This information is hidden in the decision making models of traditional systems but it can be captured and exploited in our system due to the use of argumentation. This is one of the added values of our approach.

Moreover the use of this particular argumentation framework (i.e. [8]) allows for the contextualization and the resolution of conflicts at different hierarchical levels. This case is presented in the regularity theory. As we said before, rules R_1 and R_2 are indicating that, when both decisions (i.e. "handwritten" and "printed") can be simultaneously taken, the priority is usually given to the decision "handwritten" (see that r_3 is preferred over r_1 and r_4 is preferred over r_2). However, when some specific conditions (described in the premises of rules R_3 and R_4) are satisfied, then the priority must be given to the decision "printed" (see that r_1 is preferred over r_3 and r_2 is preferred over r_4). Indeed, these conditions are based on contextual knowledge and indicate that when the height of a connected component H_{CC}, is less or equal to a threshold HT and its width W_{CC}, is greater than 20 times the threshold HT, then the decision for this connected component should be rather "printed" than "handwritten". As we will than show in section 5.3, the contextualization of the decisions allowed to sensibly improve the recognition of "printed" components (see Table 3) by the agent-based system and consists in one of the added values of our approach. So for giving the priority to the decisions based on the contextual knowledge (rules R_3 and R_4) over the decisions based on the default knowledge (rules R_1 and R_2), we use the rules C_1 (i.e. R_3 is preferred over R_1) and C_2 (i.e. R_4 is preferred over R_2) at a higher level of the theory.

With this framework we can also keep a trace of the reasoning made by the agents and more particularly we can have an explanation concerning the arguments (i.e. rules r, R and C) that have been used for making a decision and the facts (i.e. the domain knowledge) that have been considered (i.e. the values of the different parameters) as supporting information of these arguments. This allows to take into consideration one of the observations we highlighted in section 4.2 namely the need of explanation for the users.

5.2 The Agent Architecture

In this section we will describe and discuss the Linearity and Regularity Agents' architecture (see Figure 7). Indeed, the two types of agents have exactly the same architecture. Their only difference is the specific knowledge (e.g. argumentative decision theories presented above, features involved, etc.) related to the linearity and regularity points of view that instantiates the different modules of their architectural structures. Module $M2$ contains the argumentation theories

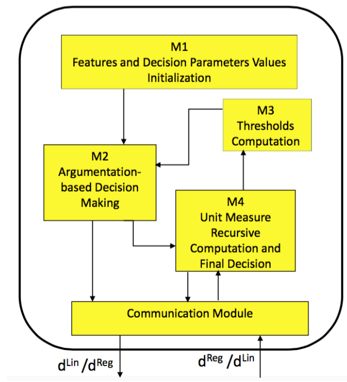


Fig. 7. Architecture of Linearity/Regularity Agent

presented above. It provides the decision (d^{Lin} or d^{Reg}) based on the information (i.e. initial values of parameters) coming from module $M1$ at the beginning of the analysis process or from module $M3$ (concerning the new values of the thresholds) after the first round of the dialogue. This decision is sent to the *communication module*. $M2$ sends also the decision to the module $M4$.

Communication modules are responsible for the communication and the implementation of the dialogue between the two agents. The communication module sends the decision to the other agent and waits for the answer. Then it informs the module $M4$.

Module $M3$ updates the decision surface in the representation space. In our case this corresponds to the computation of the new thresholds LT_1, LT_2, RT_1, RT_2 based on:

- Two parameters namely α_1, associated with LT_1 and RT_1, and α_2, associated with LT_2 and RT_2. These parameters allow to manage the shifting of the decision function and are chosen for guarantying a balance between the possible change of opinion of both agents. According to the shape of the decision functions in the baseline system and the nature of the features we are using, the shifting is not regular and that is why these two parameters have been introduced. Their values need to be fixed according to a validation set.

- The current decision i.e. "handwritten" or "printed". The decision curve is always moved towards the position of the representation point in the working space. More particularly if the current decision is "printed" then $LT_1 = LT_1 - \alpha_1$ and $LT_2 = LT_2 + \alpha_2$ for the Linearity Agent and $RT_1 = RT_1 - \alpha_1$ and $RT_2 = RT_2 - \alpha_2$ for the Regularity Agent. Otherwise, if the current decision is "handwritten" then $LT_1 = LT_1 + \alpha_1$ and $LT_2 = LT_2 - \alpha_2$ for the Linearity Agent and $RT_1 = RT_1 + \alpha_1$ and $RT_2 = RT_2 + \alpha_2$ for the Regularity Agent. This corresponds to a dynamic change of the decision functions' parameters implemented as argumentation theories.

Module $M4$ receives information from $M2$ but also the answer of the other agent transferred by the communication module. Its role is the following. It compares the two decisions and it has the following options:

- The decisions coincide. In that case $M4$ ends the decision process which provokes the end of the dialogue with an agreement between the two agents.
- The decisions are different. In that case there are two possible situations:
  1. Each decision is the same with the one taken in the previous dialogue round. In that case $M4$ communicates with $M3$ which computes the new thresholds as explained above.
  2. The two agents have simultaneously changed their decisions wrt their decisions in the previous dialogue round. Then we can have two situations:
     (a) the values of the unit measures α_1 and α_2 are greater than their minimal values in which case $M4$ decreases the values of unit measures. In the current implementation we have defined empirically the initial values of α_1 and α_2 and the decrease of their values is defined as follows: α_1=$\alpha_1 \div 5$ and α_2=$\alpha_2 \div 5$. This information is sent to $M3$. This is a recursive procedure as explained in section 4.2 ;
     (b) the values of the unit measures are lower than their minimal values in which case $M4$ ends the decision process. This provokes the end of the dialogue with a final disagreement between the two agents. In that case $M4$ returns a random decision.

The agent-based system has been implemented by using the well known agent development platform JADE [3]. However, we had also to implement an interface with SWI-Prolog [4] for running the Gorgias system that we have used for implementing the argumentation-based reasoning of the agents (i.e. module $M2$).

5.3 Experimental Results

The Datasets The aim of our experiments was to test the capability of our system to improve the final decisions proposed by the baseline system [7]. For our experiments we have chosen a public database that contains the same number of handwritten and printed words. This is the well known IAM handwriting database[5] that consists of a number of pages containing printed texts reproduced by different writers. The comparison of the two systems (i.e. the baseline system [7] and the agent-based system) has been done at the working CC level (or WCC) level. Comparison at WCC level (see Figure 1) is more appropriate than at word or pixel level, as the agents are taking decisions at that level. As IAM

[3] http://jade.tilab.com/
[4] www.swi-prolog.org
[5] http://www.iam.unibe.ch/fki/databases/iam-handwriting-database

database does not contain the ground truth (GT) (i.e. set of pairs (WCC, label)) at WCC level, we built a GT by doing a manual labeling on 25 images. These 25 pages are written by various writers. This set of images contains 3347 words and among them 1591 are handwritten and 1756 are printed. It corresponds to 2138 $WCCs$ where 1771 are handwritten and 367 are printed (see Table 2). We consider that this is the best option for evaluating the contribution of the dialogue between the two agents on the final decision (i.e. "printed" or "handwritten"). All the results concerning the 25 images are available here[6].

Evaluation In Table 2 we present the confusion matrices (i.e. number of correct and false labels wrt GT) of the baseline and agent-based system respectively. The best results for handwritten and printed $WCCs$ appear in bold. We can observe that the agent-based system has clearly improved the results concerning the handwritten $WCCs$.

Table 2. Confusion matrices for baseline [7] and agent-based system at WCC level

		GT Handwritten	GT Printed
Baseline System	Handwritten	1147	43
	Printed	624	**324**
Agent System	Handwritten	**1656**	54
	Printed	115	313
Total GT		1771	367

Indeed, as seen in Table 3, the agent-based system outperforms the baseline system as far as the handwritten $WCCs$ recognition is concerned, with a reduction of 81.67% of the error rate. Moreover, we note a real improvement of the global recognition rate as it has increased from 68.80% to 92.1%. This means that we have a reduction of 74.66% of the global error rate.

Table 3. Handwritten Recognition Rate (HRR), Printed Recognition Rate (PRR) and Global Recognition Rate (GRR) at working connected component (WCC) level.

	HRR (%)	PRR (%)	GRR (%)
Baseline System [7]	64.77	88.28	68.80
Agent System without contextualization	93.45	82.02	91.49
Agent System with contextualization	93.51	85.29	92.10
Final % of change	28.74	-3.01	23.29
Final % of change of error rate	-81.57	+25.58	-74.66

As far as the printed $WCCs$ recognition is concerned, Table 2 shows that our system is slightly dominated by the baseline system (i.e. 313/367 correct results for the agent-based system vs 324/367 for the baseline system). This is basically due to the fact that our system fails labeling as "printed" the component on the top right of the images including numbers. It is however worth noting that the

[6] http://www.math-info.univ-paris5.fr/~cloppet/PRIMA/ResultsIamPRIMA2017.pdf

agent-based system attains a score of 93.51% of correct decisions concerning the handwritten $WCCs$ while simultaneously it attains a high score (i.e. 85.29%) of correct decisions concerning the printed $WCCs$. During the processing, the agents start a dialogue only when they disagree during the initial round. In our experiments this happened in 92.32% of the cases. As shown in Table 4, the average number of dialogue rounds is 3.90, and only one dialogue out of 2138 ended up with a final disagreement (i.e. a random decision was taken). This illustrates the very good convergence of the system.

Table 4. Dialogues between Linearity and Regularity Agents

Total number of CC	Number of dialogues terminating with an agreement (right or wrong decision)	Average number of dialogue rounds
2138	2137	3.90

6 Conclusion and Future Work

In this paper we have presented an agent-based approach for solving an important problem in the field of image analysis namely the automated discrimination of handwritten/printed texts. Agent technology has already been used in the domain of image processing (see e.g. [3]) but never for dealing with this particular problem. More precisely we have shown that this problem can be transformed in a distributed decision making problem where the decision about the labeling (i.e. "handwritten" or "printed") of a text is made through a dialogue between autonomous agents. We also showed that computational argumentation (and more particularly a structured argumentation framework [8] and its associated development tool) is very well suited for implementing the decision making mechanisms of such agents. Our system can be easily extended by adding much more features (if necessary) through the insertion of additional rules in the argumentation theories. Our experimental results have proven that our solution considerably improves the performance of a (state of the art) traditional system [7]. It therefore contributes to the opening of new directions towards an increased use of agent technology (and argumentation) in the important domain of document analysis. Our system can be used in several real world applications related to this domain such as printed text detection and extraction for Optical Character Recognition (OCR), handwritten text extraction from filled up printed forms, automated detection of manually annotated printed documents. Moreover, as current electronic document management systems (EDMS) don't apply for hybrid documents (e.g. printed documents with handwritten annotations), adding an automated printed/handwritten discrimination step in these systems, would allow processing such documents in an automated way.

References

1. Belaïd, A., Santosh, K.C., Poulain D'Andecy, V.: Handwritten and printed text separation in real document. In: Proceedings of the 13. IAPR International Conference on Machine Vision Applications, MVA 2013, Kyoto, Japan, May 20-23, 2013. pp. 218–221 (2013)
2. Bench-Capon, T.J.M., Dunne, P.E.: Argumentation in artificial intelligence. Artif. Intell. 171(10-15), 619–641 (2007)
3. Bovenkamp, E.G.P., Dijkstra, J., Bosch, J.G., Reiber, J.H.C.: Multi-agent segmentation of IVUS images. Pattern Recognition 37(4), 647–663 (2004)
4. Chanda, S., Franke, K., Pal, U.: Structural handwritten and machine print classification for sparse content and arbitrary oriented document fragments. In: Proceedings of the 2010 ACM Symposium on Applied Computing. pp. 18–22. ACM, New York, NY, USA (2010)
5. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification (2nd Edition). Wiley-Interscience (2000)
6. Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artificial Intelligence 77, 321–357 (1995)
7. Hamrouni, S., Cloppet, F., Vincent, N.: Handwritten and printed text separation: Linearity and regularity assessment. In: Image Analysis and Recognition - 11th International Conference, ICIAR 2014, Vilamoura, Portugal, October 22-24, 2014, Proceedings, Part I. pp. 387–394 (2014)
8. Kakas, A., Moraitis, P.: Argumentation based decision making for autonomous agents. In: Proc. 2nd International Joint Conference on Autonomous Agents and Multi-Agents Systems. pp. 883–890 (2003)
9. Kumar, J., Prasad, R., Cao, H., Abd-Almageed, W., Doermann, D., Natarajan, P.: Shape codebook based handwritten and machine printed text zone extraction. In: Document Recognition and Retrieval. pp. 7874:1–8. San Francisco (January 2011)
10. Lins, R.D.: Meeting new challenges in document engineering. J. UCS 17(1), 1–2 (2011)
11. Pendaraki, K., Spanoudakis, N.I.: Portfolio performance and risk-based assessment of the PORTRAIT tool. Operational Research 15(3), 359–378 (2015)
12. Peng, X., Setlur, S., Govindaraju, V., Sitaram, R.: Handwritten text separation from annotated machine printed documents using markov random fields. IJDAR 16(1), 1–16 (2013)
13. Ricquebourg, Y., Raymond, C., Poirriez, B., Lemaitre, A., Coüasnon, B.: Boosting bonsai trees for handwritten/printed text discrimination. In: Document Recognition and Retrieval XXI, San Francisco, California, USA, February 5-6, 2014. pp. 902105–902105–12 (2014)
14. Saïdani, A., Echi, A., Belaïd, A.: Arabic/latin and machine-printed/handwritten word discrimination using hog-based shape descriptor. Electronic Letters on Computer Vision and Image Analysis 14(2), 1–23 (2015)
15. Wall, K., Danielsson, P.E.: A fast sequential method for polygonal approximation of digitized curves. Comput. Vision Graph. Image Process. 28(3), 220–227 (Nov 1984)
16. Zagoris, K., Pratikakis, I., Antonacopoulos, A., Gatos, B., Papamarkos, N.: Distinction between handwritten and machine-printed text based on the bag of visual words model. Pattern Recognition 47(3), 1051–1062 (2014)