Towards robust and reliable multimedia analysis through semantic integration of services

Ben De Meester, Ruben Verborgh, Pieter Pauwels, Wesley De Neve, Erik Mannens, and Rik Van de Walle

In: Multimedia Tools and Applications, 75 (22), 14019-14038, 2015.

http://link.springer.com/article/10.1007/s11042-014-2445-9

Multimed Tools Appl (2016) 75:14019–14038
DOI 10.1007/s11042-014-2445-9

Towards robust and reliable multimedia analysis through semantic integration of services

Ben De Meester · Ruben Verborgh · Pieter Pauwels · Wesley De Neve · Erik Mannens · Rik Van de Walle

Received: 5 April 2014 / Revised: 28 November 2014 / Accepted: 28 December 2014 /
Published online: 20 January 2015

Abstract Thanks to ubiquitous Web connectivity and portable multimedia devices, it has never been so easy to produce and distribute new multimedia resources such as videos, photos, and audio. This ever-increasing production leads to an information overload for consumers, which calls for efficient multimedia retrieval techniques. Multimedia resources can be efficiently retrieved using their metadata, but the multimedia analysis methods that can automatically generate this metadata are currently not reliable enough for highly diverse multimedia content. A reliable and automatic method for analyzing general multimedia content is needed. We introduce a domain-agnostic framework that annotates multimedia

B. De Meester (✉) · R. Verborgh · E. Mannens · R. Van de Walle
Ghent University - iMinds - Multimedia Lab, Gaston Crommenlaan 8 bus 201,
9050 Ledeberg-Ghent, Belgium
e-mail: ben.demeester@ugent.be

R. Verborgh
e-mail: ruben.verborgh@ugent.be

E. Mannens
e-mail: erik.mannens@ugent.be

R. Van deWalle
e-mail: rik.vandewalle@ugent.be

P. Pauwels
Ghent University - Department of Architecture and Urban Planning,
Jozef Plateaustraat 22, 9000 Ghent, Belgium
e-mail: pipauwel.pauwels@ugent.be

W. De Neve
Multimedia Lab, Ghent University – iMinds, Gaston Crommenlaan 8 bus 201,
9050 Ledeberg-Ghent, Belgium
e-mail: wesley.deneve@ugent.be

W. De Neve
Image and Video Systems Lab, KAIST, 335 Ghawak-ro (373-1 Guseong-dong),
Yuseong-gu, Daejeon 305-701, Republic of Korea
e-mail: wesley.deneve@kaist.ac.kr

resources using currently available multimedia analysis methods. By using a three-step reasoning cycle, this framework can assess and improve the quality of multimedia analysis results, by consecutively (1) combining analysis results effectively, (2) predicting which results might need improvement, and (3) invoking compatible analysis methods to retrieve new results. By using semantic descriptions for the Web services that wrap the multimedia analysis methods, compatible services can be automatically selected. By using additional semantic reasoning on these semantic descriptions, the different services can be repurposed across different use cases. We evaluated this problem-agnostic framework in the context of video face detection, and showed that it is capable of providing the best analysis results regardless of the input video. The proposed methodology can serve as a basis to build a generic multimedia annotation platform, which returns reliable results for diverse multimedia analysis problems. This allows for better metadata generation, and improves the efficient retrieval of multimedia resources.

Keywords Multimedia analysis · Reasoning cycle · Semantic reasoning · Web services

1 Introduction

Multimedia is being produced and consumed more and more each day. On the one hand, because of the ubiquitous availability of the Web through advances in telecommunication [1], and on the other hand, because of the exponential increase of portable multimedia devices (smartphones, phablets, and tablets) [24]. To harness this vast amount of multimedia resources, **efficient and effective multimedia retrieval** techniques have become a necessity [5, 10, 31]. The most widespread way of efficiently retrieving multimedia resources is by using textual annotations that accompany those resources (i.e., their *metadata*) [26]. Annotating multimedia resources manually is, however, a time-consuming and cumbersome task [8], which hence cannot cope with the current pace of multimedia production. **Automatic analysis methods** are thus needed to extract metadata from a multimedia resource.

In this article, we explain how we devised a multimedia analysis framework that can automatically annotate multimedia resources. As opposed to current related works, the proposed framework is **domain-agnostic**, and can return **reliable and robust annotations**. To create high-quality annotations, we use currently available multimedia analysis methods, wrapped in Web services.

To be able to return reliable and robust results, a **reasoning cycle** is used that consecutively (1) combines the analysis results of the already invoked analysis methods, (2) predicts which combined results might need improvement, and (3) invokes the next compatible analysis method to retrieve new results. These new results in turn start a new iteration of the reasoning cycle, until no more analysis services can be invoked. To make this framework generic, **Semantic Web technologies** are used to automatically discover the compatible Web services. By devising a problem-agnostic framework that can select and effectively combine multiple compatible multimedia analysis methods, current and future state-of-the-art algorithms can be used with minimal implementation costs, whilst maintaining robust results.

The main contributions of this paper are as follows: (1) We propose a generic approach to robustly annotate diverse multimedia resources, by using existing analysis methods, semantic technologies, and a three-step reasoning cycle; (2) We introduce a proof-of-concept that demonstrates that this approach is feasible; (3) We evaluate the implemented framework on

a video face detection use case to prove that the resulting quality of annotations is comparable or better than the current domain-specific solutions. The novelty of this work is that it can automatically assess and improve the quality of multimedia analysis results in a dynamic and generic way, whereas current composition frameworks are more static and tailored to specific use cases.

In Section 2, we review the current research efforts in this field. In Section 3, we propose our methodology of automatic Web service invocation and automatic quality assessment. This methodology is evaluated by building a proof-of-concept, which is presented in Section 4, and evaluated using a video face detection use case in Section 5. Furthermore, the same proof-of-concept is presented an optical character recognition problem, to verify that this methodology is indeed generic. Finally, there is a conclusion and future work in Section 6 and Section 7, respectively.

2 Related work

To build a robust and dynamic multimedia analysis framework using semantic reasoning, we review the current research efforts in the multimedia analysis domain (Section 2.1), and the used semantic technologies (Section 2.2).

2.1 Multimedia analysis

Recent research is conducted into improving analysis results of multimedia resources such as audio, images, and video [13, 16, 21]. Advancements are usually made by either **customizing existing solutions** for the intended use case, or by **combining multiple analysis algorithms**.

Customizing algorithms for specific use cases is mostly achieved by either using *ad hoc heuristics* to improve intermediate results [12], or by using *machine learning* to train a classifier to analyze a multimedia resource [20, 22]. This makes these high-quality multimedia analysis methods **highly specialized**, and only usable for the use case they where designed for.

Combining analysis algorithms is generally divided into two options. Either the analysis algorithms are *similar* (i.e., methods that aim to solve the same type of analysis problem, e.g., video face detection), or a *multimodal* approach is used [2]. In a multimodal approach (also called *fusion*), methods from different problem domains are combined. For example, speech recognition is used together with video face detection to improve the video face detection results. Although the approaches to fuse multiple methods are generic (e.g., linear combination of analysis results), the resulting framework is static and **needs to be trained** or uses **knowledge specific for the use case** it is intended, for example to specify the weights of the linear combination [2].

These analysis approaches continue to improve, but they usually fall short in returning highly reliable and robust results [11]. Also, by focusing on highly customized solutions, the current multimedia analysis landscape consists of many methods that are applicable to the same problem domain (e.g., person detection), but are tailored to different use cases (e.g., urban environment vs. office environment). This lowers their reliability as those methods only provide highly qualitative results in the specific use case they are intended for, and reusing them in different use cases could return results that are unexpectedly of lower quality [12]. We can conclude that there exists a high variety of static and domain-specific methods to analyze multimedia resources, but that a dynamic and generic approach is lacking.

2.2 Semantic web technologies

The Semantic Web is a layer on top of the common Web, to add machine-understandability to its content [4]. Current research is being done in making Web services machine-understandable as well, which allows for the *automatic invocation* of these Web services [9]. This automatic invocation allows for a **dynamic and automatic framework**, which negates the need of having to build custom and static multimedia analysis methods for every specific use case. Moreover, new analysis methods can be wrapped in Web services to make them automatically discoverable as well. Therefore, we can make optimal use of the already available and future analysis methods, without additional development costs.

To achieve this automatic invokability, Web services need **semantic descriptions**. These descriptions make them machine-understandable [27], which enables automatic discoverability and invokability of these services. Code Snippet 1 is an example of a very simple semantic description of a Web service, using a dummy ontology (`http://example.com/`). It states that the input type of the service should be an `ex:Image`, and that the output type of the service is a `ex:FaceRegion`. As these terms are unambiguously defined, this semantic description is enough for an automatic client to conclude that this service is an image face detection service. By describing the functionality unambiguously, invocation plans can be automatically generated. These invocation plans consist of a number of steps, each step invoking one or more Web services to collaboratively solve a given problem [28].

Code Snippet 1 Semantic description of a Web service.

```
@prefix ex: <http://example.com/>.
ex:source   a   ex:Image.        // input type
ex:output   a   ex:FaceRegion.   // output type
```

Many frameworks have been developed to produce invocation plans of Web services whilst maximizing the Quality of Service (QoS) [14, 17, 33]. These frameworks focus on maximizing QoS attributes such as response time, throughput, security, and availability. However, when using Web service composition in the multimedia analysis domain, these frameworks fall short in maximizing the effectiveness of the analysis methods, i.e., the quality of the analysis results these frameworks return.

Whereas our previous works handled automatic generation and execution of an invocation plan to solve a difficult problem by splitting it up in different steps (e.g., going from the question “Who is playing in this video” to the steps “Detect Faces” and “Recognize Faces”) [27, 28], this paper proposes a method to improve the individual steps (e.g., “Detect Faces”) by combining multiple similar existing analysis methods. In this aspect, it is orthogonal to our previous works. As a comparison to other works, a summary of the differences between related work and the proposed method is given in Table 1. The related work is compared to the proposed methodology, with respect to (1) whether the methodology is generic (i.e., can the same methodology be used in different analysis domains), (2) whether the analysis program depends on the domain (i.e., can the same analysis program be used in different domains), (3) whether it uses a combination of services to solve a given problem, and (4) what the main improvement aim is (e.g., improving the quality of the results).

Table 1 Comparison of related work and the proposed method

Related work	Generic	#Domains	Combination	Aim
Ad hoc [12, 13, 16, 21]	No	No	Maybe	Quality
Machine learning [20, 22, 29]	Yes	No	No	Quality
Fusion [2]	Yes	No	Yes	Quality
QoS combination [14, 17, 33]	Yes	No	Yes	QoS
Previous work [27, 28]	Yes	Yes	No	Execution plan
Proposed method	Yes	Yes	Yes	Quality

3 Methodology

In this paper, we devise a methodology that combines currently available analysis methods to achieve better results. In contrary with the current state-of-the-art, our framework is problem-agnostic. The methodology is inspired by a **reasoning cycle** consisting of *abduction*, *deduction*, and *induction*, as proposed by Pauwels in [19]. This reasoning cycle allows for an efficient and effective combination of the results of multimedia analysis services that are targeted to the same problem domain (Section 3.1). To make the framework generic, **semantic descriptions** of the services are used to automatically select those services that are compatible with the multimedia analysis problem at hand (Section 3.2).

3.1 Reasoning cycle

The abduction-deduction-induction reasoning cycle allows for the generation of hypotheses that can be evaluated and improved, as it allows for the construction of experience-based knowledge dynamically [19]. In this cycle, fine-grained *observations* are used to form a general *hypothesis* (abduction), which is then evaluated (deduction). This evaluation tries to discover inconsistencies within the hypothesis, and returns *predictions* of how the hypothesis should be altered to be consistent with the evaluation. These predictions guide services to collect new observations (induction), after which the previous hypothesis can be adjusted to fit the total of observations better.

Let us consider the example of video face detection, where a face of a person is detected for a large portion of a video, but not in every frame. These detections lead to the hypothesis that a certain person is visible intermittently in the given video. When this hypothesis is evaluated, it is predicted that the previously detected person should also be visible in the intermittent frames, as a person cannot suddenly disappear and reappear. Because of these targeted predictions, only the intermittent frames need to be re-analyzed. If new detections are found in a new analysis phase, the hypothesis can be adjusted, and possibly improved.

The three reasoning stages are mapped to three types of Web services (Fig. 1): analysis services, combination services, and prediction services:

- *Analysis services* are the currently available multimedia analysis methods that are wrapped in a Web service interface. These services provide for the **new observations**, as they insert new data into the framework.
- *Combination services* **combine the observations** of the analysis services and the results of the prediction services. This combination allows to form a hypothesis that can be (re-)tested by the prediction services.

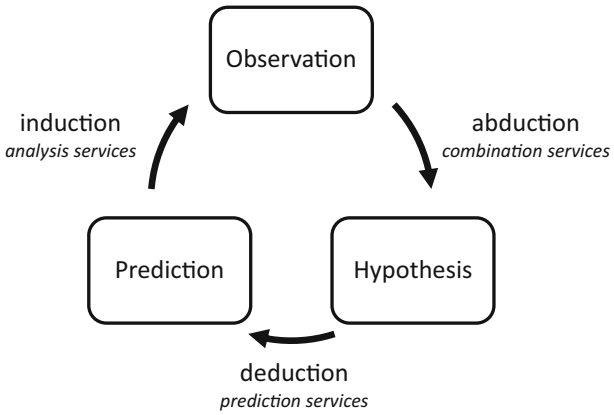

Fig. 1 Mapping abduction-deduction-induction to different types of services in a generic multimedia analysis problem solving platform

– *Prediction services* are services that **evaluate the hypothesis**. This implies that these prediction services need to contain certain domain-specific knowledge. Thus, the prediction services test the hypothesis against their contained domain knowledge for inconsistencies, and return predictions to resolve those inconsistencies.

In the next paragraphs, we elaborate a full example, to further explain each reasoning stage and how each stage is used to improve the intermediate results, using a video face detection use case.

Figure 2 shows how a video face detection request is presented to the framework, together with the input video (top of the figure). Given the request, the framework discovers two compatible analysis services, one prediction service, and one combination service. The framework iterates its reasoning cycle three times over the given video (as indicated by `(1)`, `(2)`, and `(3)`), after which a result is returned (`(4)`). Each iteration consists of an abduction step (`combine`), a deduction step (`predict`), and an induction step (`analyze`). The bounding boxes indicate which parts of the video the different Web services are invoked on. The textures indicate that the Web service returned positive results (i.e., a face was detected), and if no texture is present, the invoked Web service returned negative results (i.e., no face was detected). E.g., the analysis service of iteration `(1)` detects a face from frame 1 till frame 67, no face from frame 68 until frame 183, and again detects a face from frame 184 until frame 220. The other analysis service of iteration `(2)` is only invoked from frame 68 till frame 183, and detects a face from frame 68 until frame 121. The final result is a combination of the results of these two analysis services (`(4)`).

In short, the example starts with a first iteration, that ends with a person being detected intermittently from frame 1 till frame 67 and from frame 184 till frame 220 (Fig. 2, `(1)`). This leads to the hypothesis that a person is only visible in those frames (`combine (2)`). Prediction services mark the frames that are not conform with the domain-specific knowledge (i.e., all intermittent frames that do not contain a detection, as the domain-specific knowledge dictates that a person is a continuous object), and these frames are analyzed again (`predict (2)` and `analyze (2)`, respectively). The third iteration starts with combining the analysis results ((`combine (3)`)), after which the prediction service marks the frames that are still not conform with the fact that a person is a continuous object

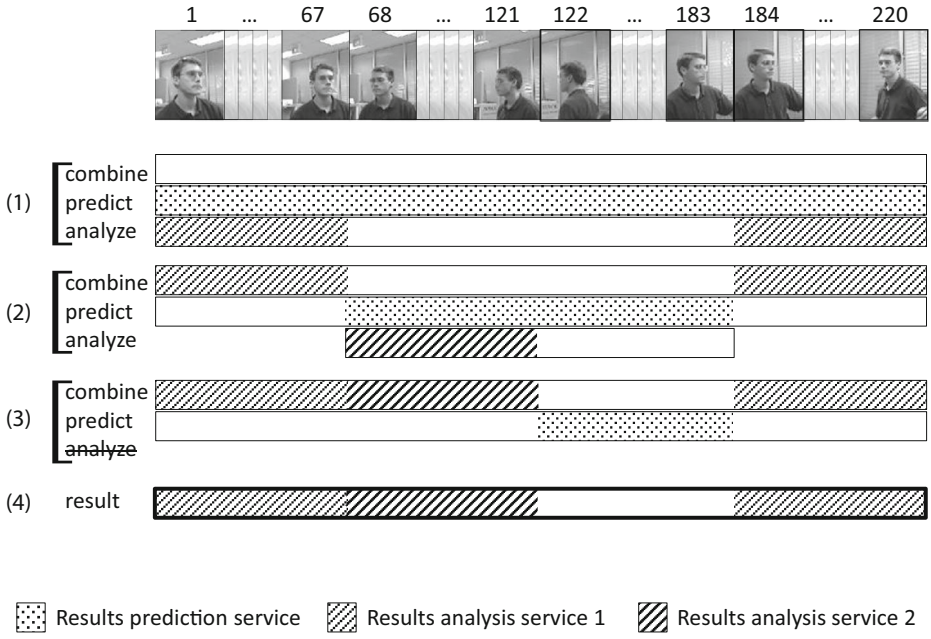

Fig. 2 Mapping abduction-deduction-induction to different types of services in a generic multimedia analysis problem solving platform. Prediction services predict where new analysis results are needed. Analysis services add new analysis results. Combination services combine the analysis results to improve the final result of the platform

(`predict (2)`). However, as no more compatible analysis services are found (`analyze (3)`), the reasoning cycle ends, and the last combination result is returned (`result (4)`). In the end, the final hypothesis is returned: a person is visible from frame 1 till frame 121, and from frame 184 till frame 220. This hypothesis is supported by the analysis results of multiple analysis algorithms and by the conformity with the domain knowledge implemented in the prediction services.

3.1.1 Abduction: Combining the analysis results

Abduction is the reasoning stage where the **most probable hypothesis is formed** based on (fine-grained) observations. With every new iteration of the reasoning cycle, new observations are added, and the hypothesis is updated to contain the biggest set of observations that are consistent with the domain knowledge implemented in the prediction services. The more observations there are, the more reliable the hypothesis can be. For example, when a face is detected in a video from frame 1 to frame 67, and from frame 184 to frame 220 (`analyze (1)`), this leads to detection observations for those frames, and non-detection observations for the other frames, which results in the hypothesis stating that the face is visible from frame 1 to frame 67 and from frame 184 to frame 220, and not visible in between (`combine (2)`). However, when another set of observations states that a face is also visible from frame 68 to frame 121 (`analyze (2)`), both sets of observations have to be combined effectively to improve the results (`combine (3)`). It could be that it is the same face that is detected, that another face is detected, or that that some of the detection results

are false. This effective combination of observations is done in the abduction stage guided by the prediction results.

In the proposed framework, this stage is mapped to **combination services**. In Fig. 2, a face region combination service combines the face regions of the first iteration with the face regions of the second iteration that coincide with the prediction results, and removes image regions of the first iteration that are refuted by the results of the second iteration [1]. The selected image regions form the new iteration of the hypothesis. It is the final version of the hypothesis that is ultimately fed back to the user who pushed the request to the platform (`result (4)`). This final hypothesis contains the combination of observations that form the strongest hypothesis, i.e. the hypothesis that is most conforming to the knowledge of the problem domain. E.g., in Fig. 2, we see that the face is predicted to be visible from frame 68 to frame 183 in order to be conforming to the domain knowledge implemented in the prediction service (`predict (2)`). The analysis results of the second iteration partially agree on that prediction (`analyze (2)`), thus the hypothesis is updated to be as conforming to the domain knowledge as possible, given the observations (`combine (3)`).

3.1.2 Deduction: Predicting the correct results

In the deduction stage, conclusions are drawn based on a certain hypothesis. These conclusion are in the form of predictions of how the hypothesis should be altered to be consistent with the problem domain. This stage evaluates the current hypothesis for inconsistencies with the problem domain, and thus tries to identify possible errors in the hypothesis. These inconsistencies are then flagged to be re-analyzed. This implies that the deduction stage **efficiently triggers the induction stage** only on those sub-problems where there is doubt of the current observations, and that its results aid the abduction stage in effectively combining possibly conflicting observations. The creation of sub-problems within a given problem is also called the divide-and conquer problem solving approach [25].

When using the initial observations of the aforementioned example of video face detection, we predict the face to be detected from frame 68 to frame 183 (`predict (2)`), as knowledge of the problem domain tells us a face cannot suddenly disappear and reappear. Analysis services that are used in the induction stage need to verify this prediction.

The **prediction services** used in the proposed methodology implement general or specific domain knowledge. Multiple prediction services can be used within one request to the platform, and one prediction service can be reused in multiple use cases (see Section 3.2).

3.1.3 Induction: Analysing the multimedia resource

In the induction stage, new observations are added to the framework. Being guided by the prediction services, analysis services are being efficiently invoked on those parts of the analysis problem that were marked to be inconsistent with the domain knowledge.

In this stage, **currently existing analysis services** are used, and the analysis results of these services are returned. Whereas most conventional multimedia analysis systems only use this inductive stage, the proposed framework employs the analysis services as a part of the reasoning cycle.

[1] For the sake of simplicity, false positives are not included in the example of Fig. 2, however, they are considered in the final platform (Section 4.2).

3.1.4 Initialization and stopping criteria

When the framework is initialized, the first iteration of the reasoning cycle is trivial (Fig. 2, `(1)`): **the hypothesis is initialized to contain no observations** (as there are none at that instant), resulting in a general prediction (i.e., *anything is possible*), after which a first analysis service is selected to do a full analysis of the original problem. The second iteration starts by trivially combining all the results of this analysis to form the first hypothesis. This hypothesis is the input of the prediction service, resulting in sub-problems that are analyzed by an alternate analysis service. These analysis results are then combined with the previous hypothesis to improve the hypothesis, leading to a third iteration of the reasoning cycle, and incrementally improving the end result.

The reasoning cycle is **stopped when all compatible analysis services have already been used on the sub-problem raised by the prediction service**. Other stopping criteria are also possible, e.g., stopping when a satisfying reliability level is achieved (Section 7). No analysis service should be reused on the same sub-problem twice (e.g., no frame should be analyzed by the same analysis method more than once), as this would insert the same result twice in the collection of observations. When the reasoning cycle cannot select analysis services anymore to provide for new observations (Fig. 2, `analyze (3)`), the result of the final hypothesis is returned to the user (`return (4)`). **Combination and prediction services can be reused**, as the added observations of the analysis services can lead to different prediction and combination results.

3.2 Service matching

The generic framework makes use of currently available services to solve the presented multimedia analysis problem. The appropriate analysis, combination, and prediction services are selected based on the **semantic description of the request**, and on the **semantic description of the available services**. For the proposed framework, this semantic description is limited to the type of input the service expects, and the type of output it returns. By knowing the input and requested output of the request, the different Web services are matched for their compatibility with the problem domain. The matching is different for analysis, combination and prediction services.

The next paragraphs explain this matching based on the applied multimedia analysis problem of video face detection. Snippet 2 shows the semantic description of the sample request, where the source of the multimedia analysis problem is a video (`ex:source a ex:Video`), and any region depicting a face visible in the individual frames of that video is requested (`ex:request a ex:FaceRegion`).

Code Snippet 2 Semantic description of the request.

```
@prefix ex: <http://example.com/>.
ex:source    a   ex:Video.        // input type
ex:request   a   ex:FaceRegion.   // output type
```

Analysis services are matched purely based on the compatibility with the input and out-
301 put types of the request. In the case of video face detection, extra semantic reasoning is done 302 by using the fact that a `ex:Video` consists of multiple resources of the type `ex:Image` to 303 select an image face detection algorithm (Snippet 1, i.e., `ex:source a ex:Image` and `ex:output a ex:FaceRegion`). Specifically, the `ex:source`

classes of the request and the analysis service need to match, and the `ex:request` class of the request needs to match the `ex:output` class of the analysis service. Using semantic reasoning, this matching may be indirect, in this case, using the fact that a `ex:Video` contains one or more `ex:Images`.

By using the semantic knowledge of the problem domain, **more general (thus more reusable) services can be selected**. A face region prediction service could be generalized to any kind of object that is visible continuously in time. This general continuous object prediction service could thus be reused for eye detection, car detection, or other detection problems. Similarly, an image region combination service instead of a face region combination service could be reused for any kind of image region combination problem.

Prediction services are selected to be compliant with the problem domain, thus they can use knowledge of that domain to evaluate the hypothesis. By **decoupling the domain knowledge of the analysis problem**, this knowledge can be reused in multiple scenarios without having to implement the domain knowledge again for every different use case. This implies that improving the prediction service for one use case potentially improves prediction results for all relevant use cases, without any extra effort, as these prediction services are selected automatically. For example, providing a better prediction service for a continuous object video detection use case automatically improves the prediction results for a video person detection use case and a video face detection use case, as the same prediction service can be reused among those different problem domains.

Using semantic descriptions of the problem domain also implies that **when the semantic descriptions of a problem domain are extended, more compatible services can be found**, leading to potentially more iterations of the reasoning cycle, and better end results. E.g., in the face detection problem domain, if we add the semantics that a face is part of a person, we can also use person detection analysis services in the face detection use case to add observations to the hypothesis. However, this implies that extra services are needed to automatically convert `ex:FaceRegion` results to `ex:PersonRegion` results and vice versa, just as an extra service was needed that converted the `ex:Video` to `ex:Images`. The matching of these extra services can be done using the same matching methods as mentioned above.

4 Proof-of-concept

We implemented the framework as a proof-of-concept, to see how significantly the proposed reasoning loop can improve analysis results, whilst remaining problem-agnostic and providing robust results across multiple given requests. The proof-of-concept consists of a *reasoning server* (Section 4.1) and a *service server* (Section 4.2).

4.1 Reasoning server

The reasoning server houses the implementation of the reasoning cycle, thus sequentially executes the abduction, deduction, and induction stages. The reasoning server is built using the *blackboard pattern* [30], storing all the intermediate and final results in a *Resource Description Framework (RDF) store* [3]. This means all observations and predictions are stored in a format compatible with the semantic descriptions of the services, which allows **reasoning to be conducted not only on the services, but also on the results**. The

semantic inference to match the services is done using the *Euler Yap Engine* (EYE) of De Roo [6].

To evaluate this proof of concept, the framework was tested on the use case of video face detection. Several videos were provided to the framework, with for each video the same semantically described request to return the face regions of the given video (see Snippet 2).

4.2 Service server

The service server provides the combination, prediction and analysis services, along with their semantic descriptions. To accommodate for the video face detection problem, several services needed to be implemented.

Two face detection services were implemented based on the Viola-Jones algorithm [29] as provided by the OpenCV initiative[2], using two different Haar classifiers to train the face detection method (`haarcascade_frontalface_alt` and `haarcascade_frontalface_alt2`). This results in two similar but non-identical face detection methods.

A simple continuous object prediction service was implemented. This prediction service looks in consecutive frames if the same object is detected (based on the vicinity of image regions across frames). If the same object is detected intermittently, interpolated image regions in the intermediate frames are flagged as possible false negatives. If an object is only detected very briefly (three frames in total or less), those image regions are flagged as possible false positives. Thus, an observation that confirms a prediction can either be a detection in case of a false negative prediction, or the absence of a detection in case of a false positive prediction.

To combine the previous hypothesis with the new analysis results, a simple image region combination service was implemented, comparing the new image regions with the already found image regions and the predicted image regions. Following the open world assumption [7], every new data from the analysis stage that is not refuted by the prediction stage is added to the hypothesis. This means:

- If a new region is found in the vicinity of a false negative prediction, that region is added to the hypothesis.
- If no region was found in the vicinity of a false positive prediction, the flagged false positive image region is removed from the hypothesis.
- A new region that is largely overlapping with a region already present in the hypothesis is combined with the already present region, resulting in a single observation.
- Every other new region is added to the hypothesis.

5 Evaluation

The proposed framework is evaluated using two use cases: one using video face detection, to show its ability to improve the results of existing methods, and one using optical character recognition, to show its generic applicability[3].

[2] http://opencv.org/

[3] Results: http://users.ugent.be/~bjdmeest/data_SemIntegrationFramework.zip

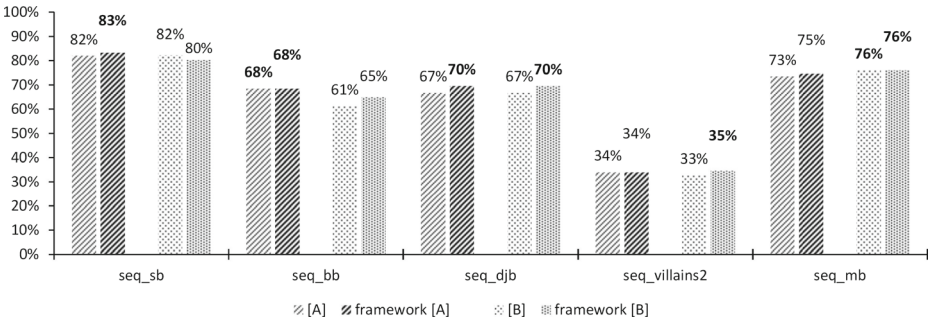

Fig. 3 F-measure of the individual analysis algorithms compared to the proposed platform for both possible invocation sequences

5.1 Video face detection

This proof of concept is evaluated using several head tracking data sets of Clemson University[4]. These data sets are provided with a ground truth and have been used by multiple head tracking experiments. The data sets contain one or more faces that are occluded, tilted and/or rotated in an office environment. A comparison is made between the results of the individual face detection algorithms and the results of the proposed framework. The best results per sequence are depicted in bold. We should note that the goal of the proposed framework is to return the best possible combination of analysis results, using the available analysis methods. This is why we will not elaborate on the effective quality of the results, but only on the gains the framework provides compared to the individual analysis methods.

Figure 3 shows the F-measure (which is the harmonic mean of the precision and recall) of the results of the framework, compared to the results of the individual methods (using Haar cascade classifier `haarcascade_frontalface_alt [A]` or `haarcascade_frontalface_alt2 [B]`). We use the F-measure as this combines precision and recall in one measure and allows for an easier comparison between methods. A division is made between which face detection algorithm was selected first in the framework, as the results can be different depending on the invocation sequence. For the data annotated with framework `[A]`, the framework started by using `[A]` for the first general analysis, and analogous for the data annotated with framework `[B]`. The best results are marked as bold.

We observe that for these data sets, the framework usually outperforms the individual algorithms. The framework performed less than `[B]` in `seq_sb` because the prediction algorithm incorrectly flagged some results as false positives, and algorithm `[A]` was not capable of detecting the faces in those frames, resulting in the wrong prediction being wrongly confirmed. Despite this one case, the framework outperforms the individual algorithms all the time. In the case of `seq_villains2`, the framework is capable of using the results of the worst performing algorithm to outperform the results of the better performing algorithm (the results of `[B]` were improved by the framework from 33 % to 35 %, whilst `[A]` remained at 34 %).

We thus observe that the framework is capable of improving the analysis results, but also that the effectiveness of the framework depends on which service is used first. This

[4] http://www.ces.clemson.edu/~stb/research/headtracker/seq/

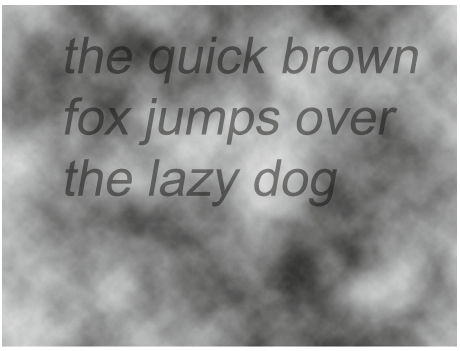

Fig. 4 Used figure for the OCR use case

phenomenon is explained by the fact that the first analysis service is used on the entire input video, whilst succeeding analysis services only need to analyze sub-problems. If the first algorithm does not perform well, there is still a lot of room for improvement by other algorithms, whilst an already qualitative algorithm cannot be greatly improved anymore. This is why certain analysis results are not improved when using the framework, as one algorithm performed absolutely better than the other in those sequences.

However, it is not possible to rank algorithms according to their performance across all requests. While we notice that `[A]` is usually better than `[B]`, this does not hold true for `seq_mb`, for example. This observation leads us to the conclusion that the proposed framework can return more robust results than the individual analysis methods, regardless of the input video.

5.2 Optical character recognition

Additionally, the framework is presented with a second problem to show the general applicability of the proposed framework: a picture containing text is presented to the framework (Fig. 4), and the contained text is requested, which is a typical Optical Character Recognition (OCR) problem.

To this end, the framework has access to two OCR services, a dictionary prediction service (i.e., a service that matches words with an internal dictionary to see whether the word exists or not), and a text combination service. The reasoning steps are presented in Snippet 3. The framework selects a first OCR service (`(1)`), flags all words that do not exist in an internal dictionary (`(2)`), selects a second OCR service to reanalyze those words (`(3)`), and combines the results (`(4)`). As no other words are flagged, the platform returns the combined result (`(4)`).

Code Snippet 3 Reasoning steps of the platform for the OCR use case

```
OCRserviceA result
             w the quick brown fdic jumps over the lazy dog Is  (1)
OCRpredictionService result
[FLAG]                         fdic                             (2)
OCRserviceB result
                               fox                              (3)
TextCombinationService result
             w the quick brown fox  jumps over the lazy dog Is  (4)
```

5.3 Discussion

The two use cases demonstrate that the framework is capable of effectively and efficiently combining currently available multimedia analysis methods in a dynamic and generic way.

Existing works about multimedia analysis and combination of analysis methods exist (e.g., fusion methods [2]), but these are static and tailored to specific use cases. Moreover, the proposed framework is capable of using these state-of-the-art methods as one of the available multimedia analysis services. This way, the framework can try to improve on the results of these state-of-the-art methods, which most of the time will result in comparable or better results than these individual methods. There also exist generic systems that, given a certain request, try to find compatible services [23], but these services are targeted to QoS metrics and do not try to effectively improve the results of the individual analysis services.

6 Conclusions

This paper described a problem-agnostic framework that can automatically select and integrate different types of Web services, to evaluate and improve multimedia analysis results. In comparison with existing methods, the proposed framework outperforms current multimedia analysis methods as it is not tailored to specific use cases, and it is capable of returning more reliable and more robust results. By making full use of currently available analysis methods, it is expected that the performance of the framework will increase as the performance of the individual analysis methods improves.

Automatic selection and reuse of Web services is achieved by matching their semantic descriptions with the semantic description of the request, where more general services can be selected by reasoning over the semantic descriptions of the services and the semantic descriptions of the problem domain. This makes the framework reusable across different problem domains with no extra implementation costs. The Web services are combined in an abduction-deduction-induction reasoning cycle, where analysis results are evaluated using prediction services and where different analysis results are combined to achieve better results.

By implementing this methodology in a proof-of-concept and evaluating this proof-of-concept for the problem domain of video face detection, we observed that this framework is capable of **automatic evaluation and improvement of analysis results**, thus providing more robust results across multiple requests. By **using prediction services to implement the problem domain-specific knowledge**, the framework itself remains problem-agnostic. Furthermore, the framework provides a degree of reliability. When multiple analysis services return the same result, the combined reliability of the result is higher than that of the individual analysis service. By reusing the framework for an optical character recognition use case, we proved its general applicability.

The evaluation showed that the framework is capable of generically improving the results of individual analysis methods in an automatic way. However, its performance does not only depend on the used analysis services, but also on the sequence of those used services. The results of this framework can be used to add relevant metadata to multimedia resources, which enables the discoverability of these resources, and facilitates their retrieval.

7 Directions for future work

The current framework returns results and their reliability on an absolute basis (i.e., detected or not detected). Extending this with *fuzzy logic* concepts [32], a future research possibility is creating a reliability measure for the used services, and for the combined reliability of the final result. This gives rise to three possible improvements to the current framework. First, the combined reliability of the returning results could be calculated in more detail. E.g., we can assume that a result which is combined from a 78 % reliable service and a 87 % reliable service has a higher reliability than the individual services, but if we want to calculate how much more reliability, we can use concepts from fuzzy logic. Second, the reliability of individual services could be computed by comparing their individual results with the combined results. The reliability of the individual services could thus be changed dynamically, and guide the reasoning process into better service selection. And third, the reliability of the intermediate results could be calculated as well. This would provide for a new stopping criterion, namely, the platform could return a result when the reliability of the intermediate results is high enough. For example, stop the reasoning cycle when the combined reliability of the intermediate results is higher than 85 %.

Other factors could also be investigated for optimal service selection, such as Quality of Service parameters [18, 34]. For example, selecting the fastest service first will improve the execution time the most, as this service will analyze the entire multimedia resource, whilst subsequent selected services will only need to analyze parts of the original problem.

Currently, the framework uses custom semantic descriptions of services to retrieve compatible web services, but there already exist ontologies and documented approaches to achieve this goal. The framework could be adjusted to use the Hydra vocabulary [15] and the RESTdesc approach [27] to enable the automatic discovery.

Acknowledgements The described research activities were funded by Ghent University, iMinds, the Institute for the Promotion of Innovation by Science and Technology in Flanders (IWT), the Fund for Scientific Research Flanders (FWO Flanders), and the European Union.

References

1. (2014). World Telecommunication/ICT Indicators database 2014, 18 edn. International Telecommunication Union (ITU)
2. Atrey PK, Hossain MA, El Saddik A, Kankanhalli MS (2010) Multimodal fusion for multimedia analysis: a survey. Multimedia systems 16(6):345–379
3. Beckett D (2004) RDF/XML syntax specification (revised). http://www.w3.org/TR/REC-rdf-syntax/. Accessed on April 15th, 2013
4. Berners-Lee T, Hendler J, Lassila O, et al. (2001) The semantic web. Sci. Am. 284(5):28–37
5. Chamin Morikawa CM, Kiyoharu Aizawa KA (2012) Iconic visual queries for face image retrieval. Journal of Convergence 3(3):39–46
6. De Roo J (2013) Euler Yet another proof Engine. http://eulersharp.sourceforge.net/. Accessed on April 7th
7. Drummond N, Shearer R (2006) The open world assumption. Presentation at eSI Workshop: The Closed World of Databases meets the Open World of the Semantic Web
8. Erdmann M, Maedche A, Schnurr HP, Staab S (2000) From manual to semi-automatic semantic annotation: About ontology-based text annotation tools. In: Proceedings of the COLING-2000 Workshop on Semantic Annotation and Intelligent Content, pp. 79–85. Association for Computational Linguistics

9. Fensel D, Bussler C (2002) Semantic web enabled web services. In: Jarke M, Lakemeyer G, Koehler J (eds) Proceedings of the 25th Annual German Conference on AI: KI 2002: Advances in Artificial Intelligence, vol 25, pp 319–319. Springer, Aachen
10. Hanani U, Shapira B, Shoval P (2001) Information filtering: Overview of issues, research and systems. User Model User-Adap Inter 11(3):203–259. doi:10.1023/A:1011196000674
11. Hauptmann AG (2005) Lessons for the future from a decade of informedia video analysis research. In: Leow WK, Lew M, Chua TS, Ma WY, Chaisorn L, Bakker E (eds) Image and Video Retrieval, Lecture Notes in Computer Science, vol 3568, pp 1–10. Springer, Berlin. doi:10.1007/11526346_1
12. Hjelmås E, Low BK (2001) Face detection: A survey. Comp Vision Image Underst 83(3):236–274. doi:10.1006/cviu.2001.0921
13. Huang YP, Lai SL (2012) Novel query-by-humming/singing method with fuzzy inference system. Journal of Convergence 3(4):1–8
14. Jaeger MC, Rojec-Goldmann G, Muhl G (2004) QoS aggregation for web service composition using workflow patterns. In: Proceedings of the Eighth IEEE International Conference on Enterprise Distributed Object Computing (EDOC), vol 8, pp 149–159. IEEE, Monterey. doi:10.1109/EDOC.2004.1342512
15. Lanthaler M, Gütl C (2013) Hydra: A Vocabulary for Hypermedia-Driven Web APIs. In: Bizer C, Heath T, Berners-Lee T, Hausenblas M, Auer S (eds) Proceedings of the WWW2013 Workshop on Linked Data on the Web (LDOW), vol 6. Rio de Janeiro, Brazil
16. Ma M, Park DW, Kim SK, An S (2012) Online recognition of handwritten korean and english characters. Journal of Information Processing Systems 8(4):653–669. doi:10.3745/JIPS.2012.8.4.653
17. Menasce DA (2004) Composing web wervices: A QoS view. IEEE Internet Computing 8(6):88–90. doi:10.1109/MIC.2004.57
18. Ohkawara T, Aikebaier A, Enokido T, Takizawa M (2012) Quorums-based replication of multimedia objects in distributed systems. Human-centric Computing and Information Sciences 2(1):11. doi:10.1186/2192-1962-2-11
19. Pauwels P, Bod R (2013) Including the power of interpretation through a simulation of Peirce's process of inquiry. Literary and Linguistic Computing (LLC) 28(3):452–460
20. Sarkar K, Nasipuri M, Ghose S (2012) Machine learning based keyphrase extraction: Comparing decision trees, naïve bayes, and artificial neural networks. Journal of Information Processing Systems 8(4):693–712. doi:10.3745/JIPS.2012.8.4.693
21. Satone M, Kharate GK (2012) Face recognition based on pca on wavelet subband of average-half-face. Journal of Information Processing Systems 8(3):483–494. doi:10.3745/JIPS.2012.8.3.483
22. Schapire RE (2003) The boosting approach to machine learning: An overview. Nonlinear Estimation and Classification. Lecture Notes in Statistics 171(7):149–172
23. Silas S, Ezra K, Blessing Rajsingh E (2012) A novel fault tolerant service selection framework for pervasive computing. Human-centric Computing and Information Sciences 2(1):5. doi:10.1186/2192-1962-2-5
24. Smith A (2013) Smartphone ownership – 2013 update. Pew Research Center: Washington DC:12
25. Smith DR (1985) The design of divide and conquer algorithms. Sci Comput Program 5(0):37–58. doi:10.1016/0167-6423(85)90003-6
26. Smith JR, Schirling P (2006) Metadata standards roundup. MultiMedia IEEE 13(2):84–88
27. Verborgh R, Steiner T, Van Deursen D, De Roo J, Van de Walle R, Gabarró Vallés J (2013) Capturing the functionality of Web services with functional descriptions. Multimedia Tools and Applications 64(2):365–387
28. Verborgh R, Van Deursen D, Mannens E, Poppe C, Van de Walle R (2012) Enabling context-aware multimedia annotation by a novel generic semantic problem-solving platform. Multimedia Tools and Applications 61(1):105–129. doi:10.1007/s11042-010-0709-6
29. Viola P, Jones M (2001) Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, pp. 511–518. Kauai, HI, USA
30. Wolfgang P (1994) Design patterns for object-oriented software development, 1 edn. Addison-Wesley (C)

31. Yen NY, Kuo SYF (2012) An intergrated approach for internet resources mining and searching. Journal of Convergence 3(2):37–44
32. Zadeh LA (1988) Fuzzy logic. Computer 21(4):83–93. doi:10.1109/2.53
33. Zeng L, Benatallah B, Ngu AH, Dumas M, Kalagnanam J, Chang H (2004) QoS-aware middleware for web services composition. IEEE Trans Softw Eng 30(5):311–327
34. Zhu Y, Jin Q (2012) An adaptively emerging mechanism for context-aware service selections regulated by feedback distributions. Human-centric Computing and Information Sciences 2(1):15. doi:10.1186/2192-1962-2-15

Ben De Meester finished his Masters degree of Computer Sciences ICT at UGent in 2013. After his graduation he started working as researcher at MMLab. His main working fields are internet tech-nologies in general and the Semantic Web in particular.

Ruben Verborgh is a researcher in semantic hypermedia at Ghent University – iMinds, Belgium, where he obtained his PhD in Computer Science in 2014. He explores the connection between Semantic Web technologies and the Web's architectural properties, with the ultimate goal of building more intelligent clients. Along the way, he became fascinated by Linked Data, REST/hypermedia, Web APIs, and related technologies. He's a co-author of two books on Linked Data, and has written several publications on Web-related topics in international journals.

Pieter Pauwels currently is a postdoctoral researcher at the Department of Architecture and Urban Planning in Ghent University (Ghent, Belgium). He holds a Master degree (2008) and PhD degree (2012) in engineering - architecture, both obtained at Ghent University. He graduated with a theoretical research on the ICT conception of an integrated architectural design environment. During his PhD research, he investigated how and to what extent information system support can be provided for architectural design thinking. After finishing his PhD thesis in 2012, Pieter started a 2-year postdoctoral research project at the Institute for Logic, Language and Computation (ILLC) in the University of Amsterdam (UvA). Currently, he is back at Ghent University, working as a full-time researcher on topics affiliated to Building Information Modelling, Linked Building Data (linkedbuildingdata.net), Linked Data in Architecture and Construction, reasoning processes involved in design thinking, semantic web technologies and the usage of semantic reasoning engines.

Wesley De Neve received the M.Sc. degree in Computer Science and the Ph.D. degree in Computer Science Engineering from Ghent University, Ghent, Belgium, in 2002 and 2007, respectively. He is currently working as a senior researcher for both the Multimedia Lab at Ghent University - iMinds in Belgium and the Image and Video Systems Lab at the Korea Advanced Institute of Science and Technology (KAIST) in South Korea. Prior to that, he spent four years working as a computer scientist in South Korea, first as a post-doctoral researcher at the Information and Communications University (ICU) and later on as a senior researcher at KAIST. His research interests and areas of publication include image and video processing (coding, annotation, retrieval, and adaptation), detection of near-duplicate video clips, face recognition, video surveillance and privacy protection, and leveraging collective knowledge for improving the semantic analysis of image and video content.

Erik Mannens is Research Manager at iMinds-MMLab since 2005 where he has successfully managed +30 projects. Since 2014 he is Professor in Semantics at Ghent University - MMLab. He received his PhD degree in Computer Science Engineering (2011) at UGent, his Masters degree in Computer Science (1995) at K.U. Leuven University, and his Masters degree in Electro-Mechanical Engineering (1992) at KAHO Ghent. Before joining iMinds-MMLab in 2005 as project manager, he was a software engineering consultant and Java architect for over a decade. His major expertise is centered around metadata modeling, semantic web technologies, broadcasting workflows, iDTV and web development in general. He is involved in several projects as senior researcher and just finished up his PhD on Semantic News Production; he is co-chair of the W3C Media Fragments Working Group and actively participating in other W3Cs semantic web standardization activities (Media Annotations, Provenance, and eGovernment). Since 2008 Erik is paving the Open Data path in Flanders. He stood at the cradle of the first Hackatons and is a founding member of the Open Knowledge Foundation. (Belgian Chapter). Since then, he is frequently invited as Open Data evangelist at national and international events. He currently actively participates in W3Cs eGov and Linked Data Platform working groups. Furthermore his team is owner of the Open Sourced Linked Open Data Publishing frameworks TheDataTank and R&Wbase. On all of these subjects he has published several papers and book chapters. He is also member of the technical committee of MTAP, ACM Multimedia, MareSO, CCNC, and SAMT. His full bio/CV can be obtained from both http://www.mmlab.be/emannens or http://www.linkedin.com/in/erikmannens. Specialties: (Linked) Open Data, Big Data Analysis, (Semantic) Web development, project management, W3C standardization (MMSEM, Video WG - Media Fragments & Media Annotation, Provenance WG), Java architect, iDTV.

Rik Van de Walle received master and PhD degrees in Engineering from Ghent University, Belgium in July 1994 and February 1998, respectively. His PhD was about Magnetic Resonance Imaging. After a post-doctoral fellowship at the University of Arizona (Tucson, USA) he returned to Ghent, became a full-time Lecturer in 2001, and founded the Multimedia Lab at the Faculty of Engineering and Architecture. Later on, Multimedia Lab became one of the founding teams of the interdisciplinary research institute iMinds. Currently, Rik's lab consists of about 45 team members. In 2004 he was appointed Full Professor, and in 2010 he became Senior Full Professor. His research interests include video coding and compression, game technology, media adaptation and delivery, multimedia information retrieval and understanding, knowledge representation and reasoning, standardization activities in the domain of multimedia applications and services. Rik is teaching several courses, dealing with fundamentals of multimedia, and (advanced) multimedia applications. These courses belong to study programs at the Faculty of Engineering and Architecture (bachelor/master of Computer Science Engineering & bachelor/master of Electrical Engineering), and the Faculty of Sciences (bachelor of Informatics). Throughout his career, Rik has been closely collaborating with a large number of research groups and partners, coming from different backgrounds: engineering, biomedical technology, sciences, arts, (digital) rights management, media and communication sciences. He has been a member/secretary/chairman of quite a large number of committees, not only within his own faculty both also outside his faculty/university. Between 2008 and 2012, he was a member of the Board of Directors at Ghent University. In 2012, he became the Dean of Ghent University's Faculty of Engineering and Architecture. Within iMinds, Rik has been leading numerous research projects, and he is acting as Head of Department of iMinds' Multimedia Technologies Research Department.