A Novel Visual Interface to Foster Innovation in Mechanical Engineering and Protect from Patent Infringement

Salvatore Sorce[1], Alessio Malizia[2], Pingfei Jiang[1], Mark Atherton[1], David Harrison[1]

[1] Brunel University London, College of Engineering, Design and Physical Sciences
[2] University of Hertfordshire, School of Creative Arts

salvatore.sorce@brunel.ac.uk

Abstract. One of the main time and money consuming tasks in the design of industrial devices and parts is the checking of possible patent infringements. Indeed, the great number of documents to be mined and the wide variety of technical language used to describe inventions are reasons why considerable amounts of time may be needed. On the other hand, the early detection of a possible patent conflict, in addition to reducing the risk of legal disputes, could stimulate a designers' creativity to overcome similarities in overlapping patents. For this reason, there are a lot of existing patent analysis systems, each with its own features and access modes. We have designed a visual interface providing an intuitive access to such systems, freeing the designers from the specific knowledge of querying languages and providing them with visual clues. We tested the interface on a framework aimed at representing mechanical engineering patents; the framework is based on a semantic database and provides patent conflict analysis for early-stage designs. The interface supports a visual query composition to obtain a list of potentially overlapping designs.

1. Introduction

Intellectual Property (IP) disputes can be resource intensive and time-consuming. A 2010 study indicated at least 24% of UK companies had experienced an IP dispute in the previous five years, damages averaging £75k-£115k were agreed in 30% of cases [1]. A more recent study found that there is a downward trend in the number of actual litigations because several disputes end in out-of-court settlement and roughly half that reach court find the patent invalid. Despite this trend, the time needed to resolve the disputes, both by litigation or settlement, remains quite long [2]. In addition, the final agreement often has a cost for both the patent incumbent and the challenger, being the result of a mediation between the two parties. On the other hand, patent applications have grown year-on-year 8.3% worldwide and 5% in Europe, with mechanical engineering accounting for 22.1% of worldwide patent applications in 2015 and maintaining an average growth of 6.4% over the preceding decade [3].

These two facts pave the way to an increasing number of disputes, due to the increasing number of patents and to the consequent increasing difficulty in data mining. Indeed, the great number of documents to be mined and the wide variety of technical language used to describe inventions, along with the different formats used to digitally represent them, represent a big obstacle for a cost and time-effective search.

In this context, an increased awareness of prior art at the beginning of a design, or during an emerging design, would help to limit these disputes. There are many available systems and tools for automated patent analysis, based on different principles and approaches: visualization, citation analysis and text mining, each using different techniques, such as Natural Language Processing, semantic analysis, property-function analysis, and so on [9]. Regardless of the approach, all patent analysis systems follow the same workflow, sketched in Figure 1. First, patents data — in form of structured representation — is stored in a database. Then, a suitable patent analysis is carried out, by means of the available querying/mining capabilities of the database.

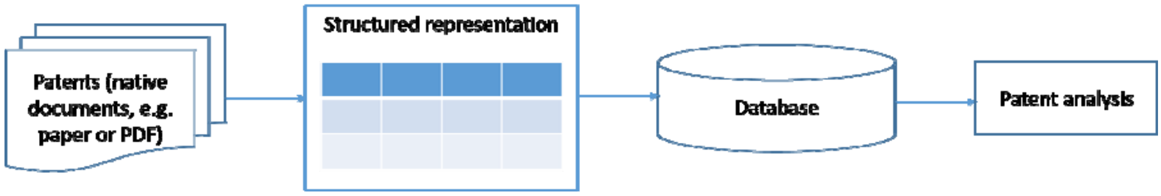

Figure 1. The patent representation and analysis workflow

Focusing on the mechanical engineering field, designers have to deal with two specific issues. One is that, while patents in such field mainly rely on graphical/visual descriptions (images, sketches), available patent analysis systems are text-based [9][18]. In these cases, the designer will typically need to enter keywords in a patent retrieval system to identify relevant prior art. A single search is rarely sufficient to capture the entire prior art. Natural Language Processing (NLP) with machine learning (e.g. IBM Watson SIIP platform [4]) has been applied to patent text search, often using statistical inference and weightings to enable text search beyond keywords. However, statistical NLP is semantically weak and is only able to predict with acceptable accuracy when trained with large datasets [5]. Commercial patent retrieval systems only employ text-based search methods and the need for advanced approaches is becoming more important as text-based techniques are increasingly challenging [6]. However, some possible approaches, such as content-based image retrieval techniques, are not well-suited to patent-images, because they mainly exploit colour images whereas patent images are mostly monochrome [7]. The requirements of a generic Patent Image Retrieval (PIR) system have been defined in the literature [8] and include a semantic-level interpretation of patents still to be developed in patent search systems [9]. Hitherto, they have been limited to a text description of images present in a patent [8]. Other PIR research has focused on image segmentation and feature-extraction but without properly capturing the semantics.

The second issue relates to detailed knowledge on query languages or patterns that normally is not part of a mechanical engineering background.

In this paper, we present a novel visual interface for helping designers to access prior art via a semantic database without requiring any specific background on query languages. The interface is designed to give visual aids for composing queries and visualising results from a patent database. It can be easily programmed to provide access to any underlying database, keeping the same layout and the corresponding affordable interaction paradigm.

In order to test the interface effectiveness, we developed our interface on top of an existing patent analysis framework in mechanical engineering. The framework is mainly composed of a domain-specific ontology stored in a semantic database that can be queried by means of the SPARQL Protocol and RDF Query Language (SPARQL) [10]. Such language, although very effective and being a de-facto standard in semantic databases, is fairly technical and not suitable for non-experts (such as mechanical designers, architects, etc.). The proposed interface allows the designers to visually compose their queries intuitively without requiring any SPARQL knowledge. It also allows for an immediate and interactive visualisation of queries results, effectively allowing designers to quickly check overlapping prior art and facilitating innovative and patentable design solutions. The interface is web-based and therefore highly interoperable including intuitive visual elements for patent analysis.

2. Background

The framework we used as a case study is composed of a patent functional representation, a domain-specific ontology and a semantic database. Figure 2 shows the relations among functional components of the framework and our interface representing the front-end.

The *functional representation* aims at expressing patents in terms of geometric features and their functional interactions. The *domain-specific ontology* enables knowledge sharing and conceptualisation, providing a standardised vocabulary for describing patented designs. The vocabulary and the relationships among geometric features and their functional interactions are encoded in a *semantic database*; this structured representation models similar working principles between an emerging design and prior art. The whole framework allows for early identification of potential conflicts and thereby can help designers steer their emerging designs away from overlapping patents.

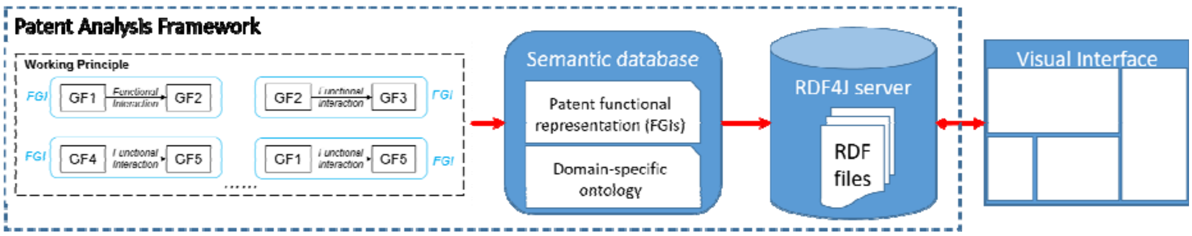

Figure 2. The patent representation and semantic querying framework

2.1. The Semantic Database

In mechanical engineering, functions are mainly realised by combinations of interrelationships between physical effects, geometric and material characteristics, known as the *working principles* [11]. Explaining the working principle in many mechanical design patents relies heavily on illustrating the Functional Interactions (FI) among the involved Geometric Features (GF). Here the structured representation of patents is obtained by means of Functional-Geometry Interactions (FGIs). They represent interacting geometric features (embodying physical effects and material characteristics) that carry a functional significance in a working principle. Both the FGIs and the semantic relations defined in the domain-specific ontology are expressed using a triple-store approach, which is a widely adopted solution for the storage and retrieval of data through semantic queries [12]. The basic form of a triple is Subject-Predicate-Object, which can be suitably used to describe GF1, FI and GF2 (Figure 2) respectively. In the framework we used, one FGI (GF1-FI-GF2) corresponds to one triple. Patent working principles, represented as FGIs, along with the domain-specific ontology are finally encoded into Resource Description Framework (RDF) format, which is a standard model for data interchange [13]. The generated RDF files are then uploaded to an RDF4J server [14], which is an open-source framework for querying and analysing RDF data. This provides access to the semantic database both through a Web interface (for browser-based access) and from an URI (Uniform Resource Identifier – for programmatic access). The server supports SPARQL queries, thus allowing the description of working principles of an emerging design in form of simple or complex queries about any existing overlap in prior art.

3. The Interface Layout

The need for different inputs other than text-based and NLP is widely discussed in literature [18]. For instance, in order to provide designers with an intuitive interface to compose queries, a visual metaphor can be used. Following a recent trend in the end-user programming community [15], block-oriented programming presents program logic as compositions of visual blocks. Tools such as Scratch, Blockly, Code.org’s lessons, and App Inventor have introduced programming and computational thinking to a huge audience, reaching people of all ages and backgrounds. Supporting a block-oriented metaphor, we aim at easing the designer's cognitive load in formulating queries and freeing him from the burden of learning SPARQL language. Query results are visualised including some interactive feature (such as a thumbnail-preview to allow a quick navigation of the patents ranking).

Figure 3 shows the interface layout organised right-left and top-down, following the standard F-shaped pattern of reading [16]. With reference to Figure 3, the title bar (area #1) shows the title "D4i" (Design for Invention), and three option buttons. The "Viewer" button activates the design viewer that in our case is a 3D rendering of the design in area #2. The "Simple" button hides the design view, thus making more room for the other areas and items in our interface. Indeed, a designer may need to check for patent infringements before starting any new design. For such reason, the 3D view can be closed, thus making our interface also suitable in cases where a 3D view is not applicable/available. The "Guide" button shows a short video guide on how to use the tool to improve usability.

The area #2, when activated, shows the design viewer. It is worth noting that, since we are considering the specific domain of mechanical engineering, the design viewer shows a 3D design, along with common features for view manipulation (rotate, pan, zoom and explode to show the design components). The visual interface is also able to import any 3D design in OBJ format (a standard file format for 3D objects to support interoperability with existing CAD software). The area #3 shows the query result — a list of existing patents overlapping with the proposed design; the visualisation of the list shows the number of patents matching the design and can be reordered; a single click on an item activates a preview of the patent and a double-click opens the corresponding file. The area #4 shows query block-oriented commands available for querying the database. Each block can be drag-and-dropped in the area #5 to compose a query. Finally, area #6 shows a preview of the selected document.

3.1. Interface Features and Functions

Our use case focus on an example of a typical workflow for searching a given FGI (Functional-Geometry Interaction), and in particular for searching products having a hole in a plate.

The designer starts composing a query by picking the appropriate functional blocks from the "Available blocks" area (#4 in Figure 3) and drags those to the "Query composition" area (#5 in Figure 3). Each query starts with the "Search for products" block, and all other blocks must be wedged after it. The block shapes suggest possible ways to compose a query and prevent from wedging meaningless or wrong blocks to queries, following the poka-yoke principle [17]. According to the triple-based structure of our semantic database, each FGI is composed of a Geometric Feature GF, then a Functional Interaction FI, and then another GF. In this case, a query to search for a single FGI is composed by drag-and-drop of a GF (orange) block under the "Search for products" (green) block. Then a FI (purple) block is wedged to the first GF, and last another GF block is placed at the end of the line. This block arrangement semantically composes the statement: "Search for products with the geometric feature GF1 functionally interacting by means of FI with the geometric feature GF2", i.e. a triplet.

Once a block is correctly placed, a dropdown list allows the user to select a specific item (a GF or a FI, depending on block type). In our example, we selected "Hole", "Locate at", and "Plate" respectively. The dropdown lists are populated at the start, by querying the database for the available GFs and FIs.

In order to run a query, the user must click on the green disc with a triangular symbol, the "Play" button. This will start the background process to translate the visual query to its textual version in SPARQL and send it to the server that stores the semantic database, and returns the results. This translation step (from blocks to SPARQL) can be adapted to different target output languages, thus making our interface adaptable to different patent analysis systems. The matching results are listed in the list box on the right side of the interface. Figure 3 shows the interface status at the end of this process. The results list is interactive, and in particular, it allows users to see a preview of a given product within a patent by clicking on the corresponding name in the list. The query may be saved by clicking on the blue disc with a downward arrow, the "download" button below the "Query composition" area.

It is worth noting that more complex or detailed queries can be composed by adding block lines below the first one, following the example above. Furthermore, simpler queries can be composed by adding fewer blocks, for instance, supporting a single GF. In this case, the tool will search for products having that GF, leading to a (possibly) longer list of results. Last, it is not mandatory to choose an item from the dropdown lists. In this case, the search will be carried out as if there is a "*" wildcard in the corresponding SPARQL field, meaning "search for all". Finally, the two available blocks "Function action" and "Function object" allow for database-specific search. Indeed, the databased used in this case study included additional features coded to describe a product or a patent itself. These features allow for generic search, which can turn out to be useful especially at the beginning of a design process.

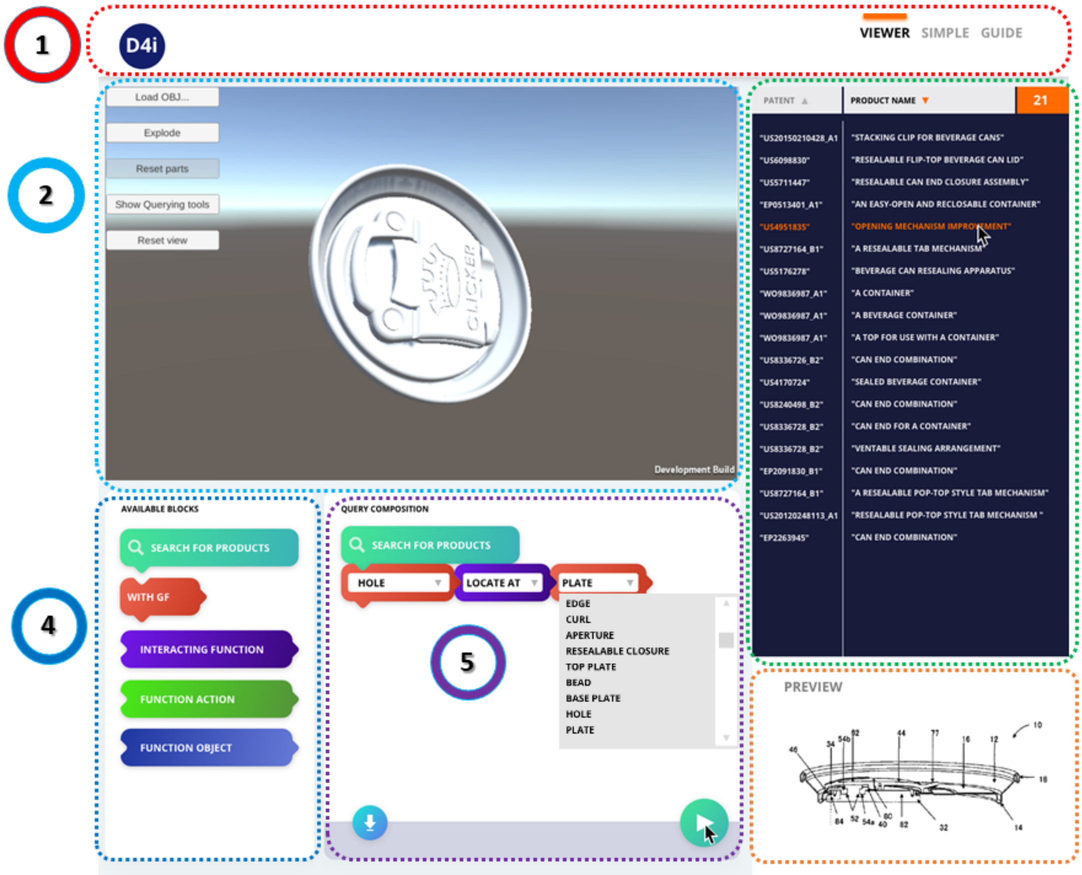

Figure 3. The interface layout

4. Discussion & Future Work

As a first preliminary evaluation, here we briefly report an informal and qualitative testing, conducted on three colleagues skilled in mechanical design. The test was conducted individually, each session took 10 minutes at most, and none of the participants had seen the interface before. The current layout and shape of the visual items have been perceived as suitable and intuitive, including the position and size of the interactive 3D viewport. All three participants reported that the interface affords the drag-and-drop action for the blocks, and the drop-down lists suggest to choose one item among the available ones. Next, the meaning of interface symbols, terms and acronyms need to be clearly defined. This is an expected result, since we used our interface to access a specific database but it does not affect the interface effectiveness. Lastly, we observed some uncertainty about how and when to visualise the results of the query in terms of the role of the green "Play" button in the "Query composition" area was not so clear; and there was no clear understanding of when a query could be executed depending on the number and type of blocks added. This suggests us to implement a "continuous" view of the results, so that every time a change occurs in the "Query composition" area, the "Results" area should show the corresponding results. This will require one less step of interaction (the click on the "Play" button), and should make it more evident that adding blocks will lead to a refined list of results.

We are currently working on the definition of a possible model of the "state" reached in the patent checking process. This would allow designers to save snapshots of their work at given steps, and to compare them in order to check the effectiveness of their design choices. Furthermore, we are currently planning a thorough evaluation, involving people from the design engineering profession.

In this paper, we presented a visual interface supporting an easy formulation of semantic queries for early detection of possible patent infringements. The interface is based on the well-established principle of "block-oriented programming". The visual interface highlights the potential prior art conflicts and areas of innovation of the emerging design in the patent space. As a desirable side-effect, the early detection of existing relevant prior art should foster design creativity, suggesting how to achieve the same functions with different operating principles or geometric features. Despite our case study being mechanical design, the proposed interface may be adapted to other functional representations in different fields where intellectual property protection and development is a relevant issue.

References

[1] Greenhalgh, C., et al (2010), "Intellectual Property Enforcement in Smaller UK Firms: a report for the Strategic Advisory Board for Intellectual Property Policy (SABIP)."

[2] Jeon, H., (2015), Patent infringement, litigation, and settlement, In Economic Modelling, Volume 51, 2015, Pages 99-111, ISSN 0264-9993, https://doi.org/10.1016/j.econmod.2015.07.019.

[3] World Intellectual Property Organization (WIPO), "World Intellectual Property Indicators 2017", http://www.wipo.int/publications/en/details.jsp?id=4234

[4] http://researcher.watson.ibm.com/researcher/view_group.php?id=2134

[5] Cambria, E., White, B., (2014), "Jumping NLP Curves: A Review of Natural Language Processing Research [Review Article]," in IEEE Computational Intelligence Magazine, vol. 9, no. 2, pp. 48-57, May 2014, doi: 10.1109/MCI.2014.2307227

[6] Bhatti, N., Hanbury, A. (2013), "Image search in patents: A review", International Journal of Document Analysis and Recognition (IJDAR), vol. 16 issue 4, Dec. 2013, pp. 309–329.

[7] Vrochidis, S., et al (2010), "Towards content-based patent image retrieval: A framework perspective", World Patent Information, vol. 32 issue 2, Jun. 2010, pp. 94–106.

[8] Vrochidis, S., et al (2012), "Concept-based patent image retrieval", World Patent Information, vol. 34 issue 4, Dec. 2012, pp. 292–303.

[9] Abbas, A., et al (2014), "A literature review on the state-of-the-art in patent analysis", World Patent Information, vol. 37 issue 4, Jun. 2014, pp. 3–13.

[10] Jiang, P., Atherton, M., Harrison, D. Malizia, A., (2017) "Framework of mechanical design knowledge representations for avoiding patent infringement", In: Proceedings of the 21st International Conference on Engineering Design (ICED17), Vol. 6: Design Information and Knowledge, Vancouver, Canada, 21.-25.08.2017.

[11] Pahl, G., Beitz, W., (2006), Engineering Design: A Systematic Approach. Third. Springer-Verlag London. doi:10.1007/978-1-84628-319-2.

[12] Rusher, J., (2003), "TripleStore." In Semantic Web Advanced Development for Europe (SWAD-Europe), Workshop on Semantic Web Storage and Retrieval.

[13] Atherton, M., Jiang, P., Harrison, D. and Malizia, A. (2017), "Design for invention: annotation of Functional Geometry Interaction for representing novel working principles". Research in Engineering Design. ISSN: 0934-9839.

[14] RDF4J "Eclipse RDF4J – a Java Framework for RDF", http://rdf4j.org/

[15] "Foreword," presented at the Blocks and Beyond Workshop (Blocks and Beyond), IEEE, 2015

[16] Nielsen, J., (2006), "F-Shaped Pattern For Reading Web Content", https://www.nngroup.com/articles/f-shaped-pattern-reading-web-content-discovered/

[17] Nikkan Kogyo Shimbun (1988). Poka-yoke: improving product quality by preventing defects. Productivity Press. p. 111. ISBN 978-0-915299-31-7

[18] Bonino, D., Ciaramella, A., Corno, F. (2010). "Review of the state-of-the-art in patent information and forthcoming evolutions in intelligent patent informatics", World Patent Information, 32(1), 30-3