



HAL
open science

IntuiSketch, a pen-based tutoring system for anatomy sketch learning

Islam Barchouch, Omar Krichen, Eric Anquetil, Nathalie Girard

► **To cite this version:**

Islam Barchouch, Omar Krichen, Eric Anquetil, Nathalie Girard. IntuiSketch, a pen-based tutoring system for anatomy sketch learning. International Graphonomics Society 2023, Oct 2023, Evora, Portugal. hal-04195461

HAL Id: hal-04195461

<https://inria.hal.science/hal-04195461>

Submitted on 4 Sep 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Copyright

IntuiSketch, a pen-based tutoring system for anatomy sketch learning

Islam Barchouch, Omar Krichen, Eric Anquetil, and Nathalie Girard

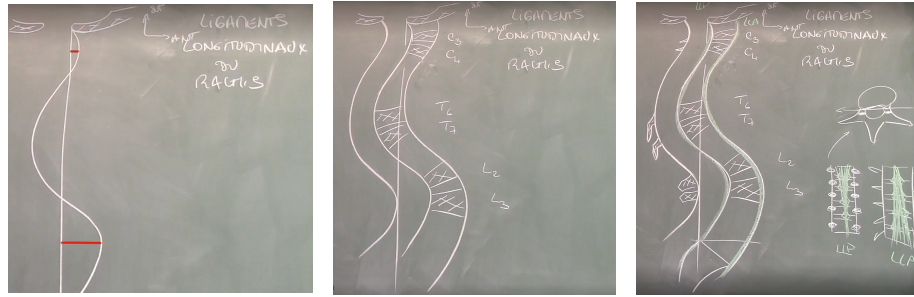
Univ Rennes, CNRS, IRISA
FirstName.LastName@irisa.fr

Abstract. This paper presents the first works on IntuiSketch, a pen-based intelligent tutoring system for anatomy courses in higher education. Pen-based tablets offer the possibility to have pen and touch interaction, which mimics the traditional pen and paper setting. The objective here is to combine online recognition techniques, that enable to interpret the sketches drawn by the students, with tutoring techniques, that model the domain knowledge. IntuiSketch is able to analyze the student drawings relatively to a problem defined by the teacher, and generate corrective feedback. The online recognition is based on the bi-dimensional grammar CD-CMG (Context Driven Constraint Multiset Grammar) which models the document structure, coupled with a fuzzy incremental classifier, which is able to learn from few examples. The tutoring system is based on constraint modeling, which enables to define domain and problem knowledge, and to analyse the student production relatively to the constraints that have to be satisfied to solve the problem. In this paper, we present a new architecture for anatomy sketch targeted intelligent tutoring system that combines different techniques. We also present a qualitative study of the feedback that our first system version is able to generate on a case study.

Keywords: Online sketch recognition · Bi-dimensional grammar · Constraint-based tutors · Digital learning · Generative drawing

1 Introduction

This work is part of the SKETCH project, which aims to design an intelligent tutoring system for anatomy courses in higher education. Fig. 1 illustrates an anatomy sketch used as source material in class. The pedagogical foundation of this work lies in the concept of generative drawing [6], which stipulates that learning by drawing can enhance student performance and understanding of the course material. We are therefore interested in problem solving by drawing anatomy sketches that satisfy the constraints defined in the instruction. Another pedagogical foundation of this work is the importance of generating prompt feedback adapted to the problem-solving process followed by each student. In education psychology, there are generally two types of feedback: delayed, and prompt feedback. According to [8], real-time feedback is more effective for



(a) First drawing step of the spine (b) Intermediate drawing step of the spine (c) Last drawing step of the spine

Fig. 1: Example of the different steps of drawing a complete spine, created by a teacher in class

student learning, since it enables him/her to quickly detect errors and correct them.

The objective of our work is to design a pen-based tutoring system based on these pedagogical principles. To ensure an intuitive user experience, our system has to allow free-hand sketching, *i.e.* there is no need to use drag-and-drop techniques to compose an anatomy sketch.

In this context of freehand drawing, our IntuiSketch system must be able to interpret these so-called semi-structured drawings. That is to say, the interpretation is not only in terms of their shape, but also in terms of their structure, *i.e.* spatial and geometric relations linking each element in the drawing. In this work, we use a combination of syntactic ([13], [3]) and statistical ([12], [11]) approaches for the online interpretation of student productions. Moreover, the interpretation is not only online, but also *eager*, *i.e.* strokes are interpreted in an incremental manner (stroke by stroke). This eager interpretation mode allows the generation of prompt feedback. The syntactic approach consists in using a bi-dimensional grammar CD-CMG (Context Driven Constraint Multiset Grammar) [9], which enables to model the document structure with composition rules. The statistical approach uses a fuzzy inference system named Evolve [2], which is able to learn from few examples. This capacity is important in our context, since our aim is to let the teacher define, by drawing them, new anatomy sketch exercises *via* an author mode. Therefore, new classes of anatomy objects are learned by the system each time the teacher introduces a new concept. This is one of the reasons we did not consider deep learning techniques in our work, even if these techniques have made great strides recently and are the state of the art in terms of graphics and handwriting recognition [10, 4]. Another reason is, even though deep learning techniques are the best suited to recognize isolated and complex shapes, that are not able to model the document structure in a fine explainable way that could be translated, or used to generate, corrective and adaptive feedback to the students. This recognition part of the system is encapsulated in a

recognition engine. This paper will focus on the tutoring aspect of the system and its interaction with the recognition engine. A tutoring system is, as defined in [1], a "computer program that uses AI techniques to provide intelligent tutors that know what they teach, whom they teach, and how to teach". There are two major approaches for knowledge modeling for tutoring systems: *cognitive tutors* and *constraint based tutors*. Cognitive tutors model the knowledge in terms of expert rules. The basic principle in place here is that each step of the student solution is compared to an ideal step modeled by a chain of expert rules. If a match is found, the step is valid, or else a corrective feedback is generated. These kind of systems are costly to build, especially for ill-defined domains like anatomy sketching, which are not characterized by a formal theory that can be defined beforehand (contrary to other domains such as geometry [7] or mechanics [3]). Constraint-based tutors take a simpler approach, that each problem, and each step, can be defined by a set of constraints that must be satisfied. The analysis of student's production step by step then consists in checking that no constraint defined for the current step is violated. If there is, a corrective feedback is generated in relation to the error made by the student. This kind of modeling is more suited to our field.

In this work, we propose a hybrid architecture for a sketching-based tutoring system to finely analyze student's productions and provide relevant feedback in the context of anatomy courses. This architecture is based on the one proposed in IntuiGeo [7], a pen-based tutor for learning geometry. Our IntuiSketch project focuses on the composition of anatomical structures, which presents a higher degree of freedom compared to the drawing of geometric figures. As a result, we are faced with more challenging tasks, both in terms of interpreting such semi-structured sketches, as well as the fine analysis of these production relatively to a defined problem constraints.

Indeed, if the geometric constraints are sufficient to model the knowledge of the domain in the context of IntuiGeo, for anatomy drawings this is not the case, we therefore propose here to combine different types of information: shape, structure, position, in addition to certain geometric constraints. This enables the system to tackle the more challenging task of analyzing semi-structured patterns.

This paper is organized as follows. Sec. 2 presents the interpretation approach based on CD-CMG grammar and Evolve classifier. The tutoring system architecture is presented in Sec. 3. The interaction between constraint based modeling and pattern recognition is detailed in Sec. 4. First (preliminary) qualitative experiments on a defined use case (*i.e.* anatomy problem) will show the feedback typology in Sec. 5. Conclusion and future works are given in Sec. 6.

2 Recognition engine

The 2D recognition engine is at the core of the IntuiSketch intelligent tutoring system, and is based on the Context Driven Constraint Multiset Grammar (CD-CMG) [7] for the structural modeling of the document and the incremental classifier Evolve [2] for the statistical recognition of anatomical shapes.

2.1 CD-CMG principles

The structural aspect of CD-CMG ensures that the recognition process takes into account the surrounding context of each element. This context can include the relative position of other elements, their semantic associations, and the overall structure of the sketch. The CD-CMG formalism is a set of production rules that describe the syntax and semantics of elements in the overall sketch. As illustrated in Fig. 2 (detailed thereafter), the production rules of this formalism define the process of replacing one multiset of elements by another, *i.e.* creating new elements by specifying the conditions under which this replacement can take place. By incorporating contextual constraints, the grammar ensures that the recognized elements respect the structures expected in the specific domain. It combines the concepts of contextual analysis, constraint satisfaction and multiset operations to capture the complexity and variability of sketches.

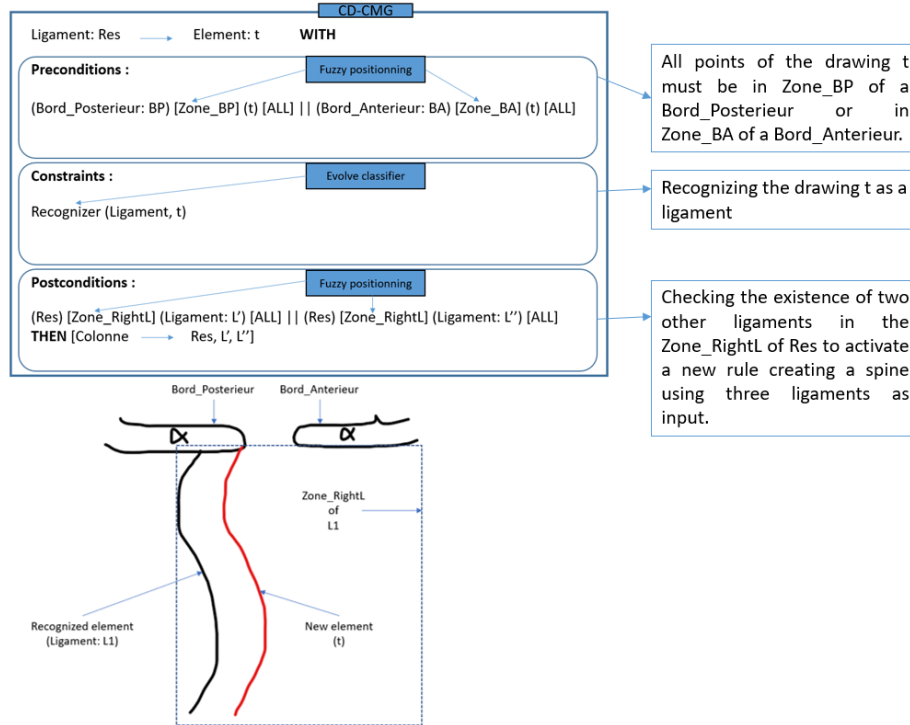


Fig. 2: One CD-CMG production rule representing an intermediate step to create a ligament (in red) in the spine example

By applying these rules iteratively, the recognition engine can analyze and recognize the various elements of the document.

In CD-CMG, preconditions, constraints and postconditions are an integral part of grammar rules. Fig. 2 represents the intermediate step of the example mentioned in Fig. 1b, the CD-CMG production rule for the creation of a ligament takes as input an elementary stroke t . Firstly, the preconditions in the CD-CMG grammar rule define the requirements that must be satisfied before the rule can be applied, to check the rule applicability. Contextual consistency is confirmed by checking that the trace t belongs to defined zones of the document, using fuzzy positioning [5] which uses fuzzy logic to process uncertain spatial information. It represents spatial information using fuzzy inference systems to determine the degree of certainty of an element's position. This approach, according to [5], enables flexible and robust processing of imprecise or ambiguous spatial relationships. Next come the constraints used to enforce consistency, as well as structural or semantic coherence within the grammar. This enables to recognize the shape of the pattern and check the structural constraints that allow to recognize the pattern as a ligament, using the Evolve classifier. Finally, postconditions are used to define the changes or transformations that occur in the structure or attributes of elements after the rule has been applied. They define the transformation or replacement that takes place in the document structure so that new elements are predicted and the document is updated.

The CD-CMG mechanism allows domain-specific knowledge and problem constraints to be modeled. These constraints define the expected properties and relationships between sketch elements, ensuring that the elements recognized comply to the principles and rules of anatomy provided by the teacher in author mode. By checking these constraints, the recognition engine guarantees the validity and consistency of the sketches interpreted in student mode.

2.2 Incremental classifier Evolve

To improve the recognition process, we use the Evolve classifier [2], which is capable of learning from a few examples. Indeed, some patterns cannot be modeled by the CD-CMG grammar like those shown in Fig. 3. Such patterns are characterized by a high degree of drawing liberty and require the integration of a classifier in order to recognize them. Evolve is well suited to the dynamic nature of sketch recognition, as it adapts and evolves according to the nature of the sketches encountered and the annotations associated with it, as defined in [2]. The Evolve classifier adapts its own internal parameters and updates its knowledge representation based on new examples and their associated labels.

The recognition engine is designed to analyze user's input in real time, taking into account what pen-based tablets have to offer in terms of interaction. As students draw their anatomical sketches, the recognition engine continuously analyzes the features and uses CD-CMG grammar to recognize and classify the different elements present in the document. Once the user's input is recognized, the tutoring aspect of the system takes place by analyzing the validity of the drawing relatively to the defined problem and the generation of feedback that is adapted to the student resolution state. We present in the next section the new tutoring architecture of IntuiSketch.

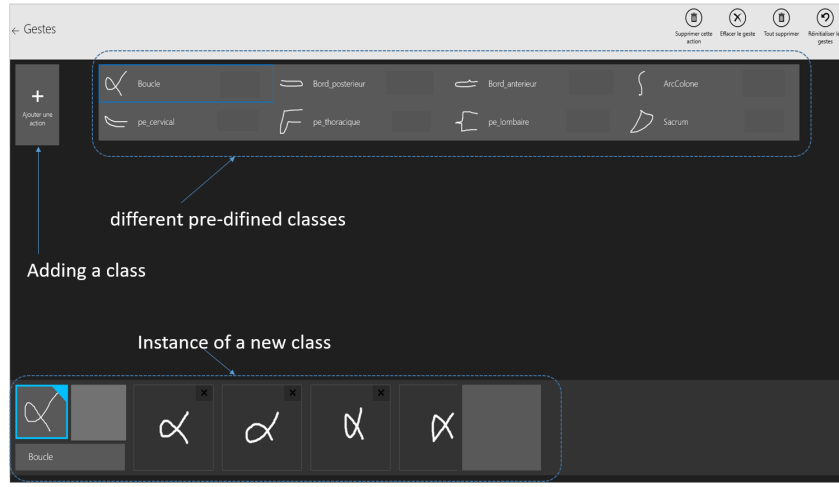


Fig. 3: Class set of drawings defined through examples drawn by the user in Evolve classifier

3 IntuiSketch architecture

3.1 Architecture founding principles

The IntuiSketch system is based on a new architecture (presented in Fig. 4), which builds on the IntuiGeo tutoring system [7]. It is composed of a *recognition engine* which interprets the sketches drawn by the user on the basis of structural and statistical knowledge. The *authoring module* enables teachers to create exercises by drawing a solution example. The *domain module* represents the domain declarative knowledge. Since it is inspired by the Constraint Based Model paradigm, our domain model encapsulates, in the form of a knowledge graph, the constraints that a correct student solution must satisfy, given the teacher reference sketch.

The learner module (known also as student model in the literature) uses the knowledge graph (K.G.) and the recognition engine to check each step of the student resolution. Each drawn stroke is either recognized as an element by the recognition engine or rejected (if it does not correspond to any of the defined classes). The result of the interpretation is then matched with the knowledge graph to check the satisfied and unsatisfied constraints. There are different types of constraints: geometric, spatial zoning, and shape. Geometric constraints are directly checked within the knowledge graph while spatial zoning and shape constraints require the learner module to interact with the recognition engine (namely Evolve and the zoning/positional algorithms of the CD-CMG grammar) to verify them. If one or more constraints are not satisfied, a corrective feedback is generated for the student.

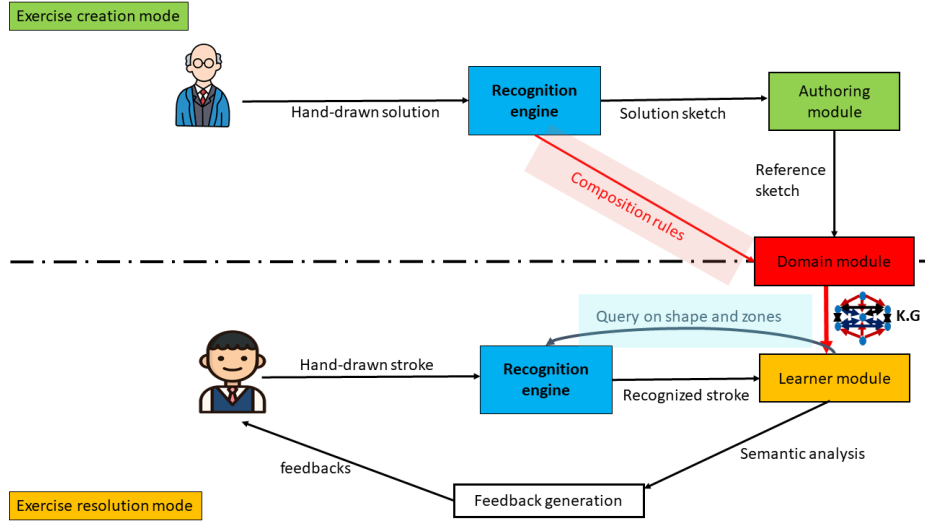


Fig. 4: IntuiSketch tutoring system architecture

3.2 Adaptation of the architecture for complex sketches

The novelty of this approach, compared to IntuiGeo, is the evolving nature of the recognition engine and the domain module. Since Geometry is a well-defined domain, the knowledge base of IntuiGeo was static, *i.e.* defined by 2D euclidean geometry theorems. By contrast, anatomy sketching is an ill-defined domain, which means that is impossible to code a priori the domain knowledge. Therefore, this new domain knowledge has to be *evolving*, *i.e.* be able to acquire new knowledge from each new exercise created by the teacher that introduces new anatomy concepts. By the same token, the recognition engine has to be evolving too, *i.e.* able to learn new anatomic shapes, as well as new composition rules, if the teacher introduces unseen elements in his/her drawing. Here the domain knowledge will be enriched directly from the evolving recognition engine, with new composition rules feeding the domain module each time a new concept is introduced by the teacher in his/her drawings (red arrow in Fig. 4).

Another important improvement in this new architecture is the interaction between tutoring and pattern recognition. In the geometry domain, the learner module just checks the geometric constraints validity, there is therefore no need for an interaction with the recognition engine. In this new domain, there are new types of constraints (shape and zoning) which mean the learner module and the recognition engine are in constant dialogue to check constraint satisfaction (gray arrow in Fig. 4).

In this paper, we present first works on this new architecture. We focus on a use case problem (a spine exercise), with a priori defined composition rules and shapes classes (*cf.* Sec. 2), as well as the knowledge base of the domain model. Automatizing these processes, by putting the teacher in the loop, will be the

subject of future works. Details on the interaction between the tutoring aspect of our system and the recognition aspect are given in the following section.

4 Interaction between tutoring and pattern recognition

The interaction between tutoring and recognition takes place in the domain module which represents the domain declarative knowledge. This interaction takes also place in the learner module, which is responsible of analyzing the resolution state of the student.

4.1 Domain module: Knowledge graph construction

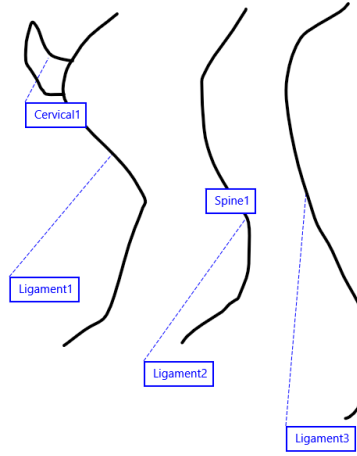


Fig. 5: Part of spine solution sketch

Fig. 5 illustrates part of a spine's drawing, including three constituting ligaments, the structure of the spine, and a cervical element. As shown in Sec. 2, all these elements are linked by spatial zoning relations (fuzzy positioning) and mathematical constraints (*e.g.* parallelism). An element is also defined by its shape, which can be described structurally (a spine is composed of three parallel ligaments) or described statistically with Evolve, classifying the stroke as one of the defined anatomy classes. These constraints are fed, in the domain module, to a knowledge graph which represents the objects present in the document and their relations, as illustrated in Fig. 6.

The spine rule is *purely structural*, *i.e.* the spine depends only on three ligaments that are parallel and close together in space. Therefore, in the K.G., we consider the spine as a child of its three components. The geometric constraints

(in red in Fig. 6) are propagated from the spine rule to the ligament nodes, thus creating links between these nodes (two directional arrows). Self directional arrows (light blue) represent the reflexive constraints, directly extracted from the CD-CMG production constraints block (see *e.g.* Fig. 7).

According to the cervical production rule presented in Fig. 8, the cervical composition depends on the presence of a spine, and its relative position with respect to the said spine, this position being defined in the preconditions of the grammatical rule. This creates a link from the cervical node to the spine node containing the zoning constraint, extracted from the production preconditions bloc (*cf.* Fig. 6).

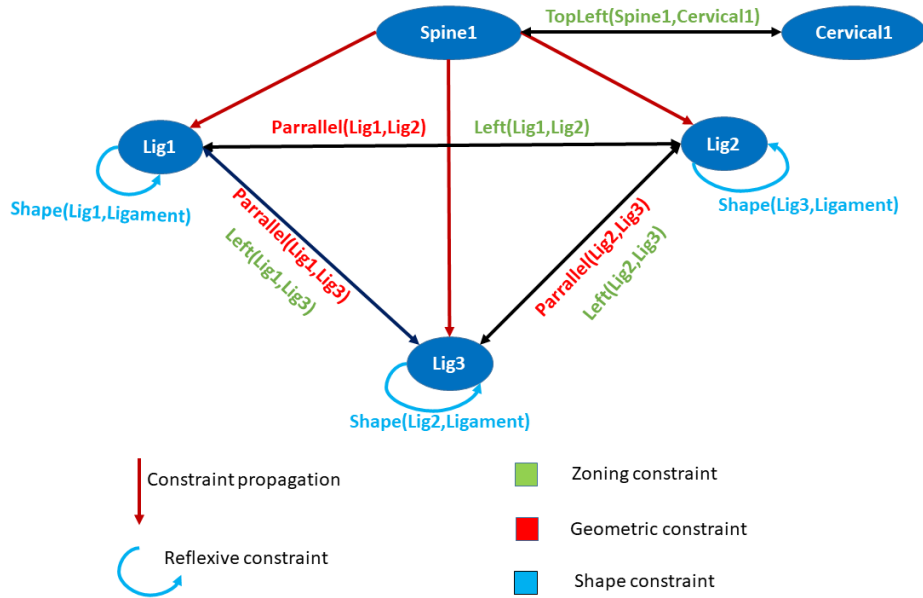


Fig. 6: Part of the knowledge graph for the spine example

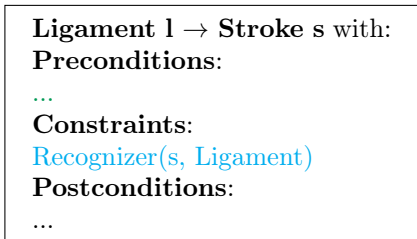


Fig. 7: Ligament composition rule

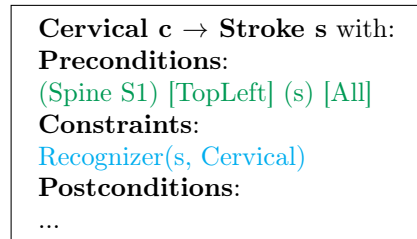


Fig. 8: Cervical composition rule

From this example, we can see that the knowledge graph representation is a kind of *conversion from a grammatical representation of the document to a graphical representation of the document*. This knowledge representation enables constraint-based modeling of the student’s resolution process, by matching his/her actions with the nodes of the graph, as well as verifying the validity of his/her solutions against the problem constraints.

4.2 Learner module: matching and feedback generation

In the exercise resolution mode, the problem and its constraints are represented by the knowledge graph. We can say that we are in the context of constraint-based modeling of the student’s solution since we formulate the sketching exercise in terms of a set of anatomy objects (nodes) and the constraints that connect them (arrows), all of which must be satisfied for the problem to be solved.

Each student action, *i.e.* new stroke, is first analysed by the recognition engine (*cf.* Fig. 4). And, the resulting recognized element (or elements) is matched with one of the knowledge graph nodes. The matching principle is based on the minimization of a constraint score defined as follows.

Definition 1 (Constraint score).

The constraint score of a node n when matched with a new element E_{new} is:

$$Score(E_{new}, n) = \frac{CGeom(E_{new}, n) + CZones(E_{new}, n) + CShape(E_{new}, n)}{|CGeom(n)| + |CZones(n)| + |CShape(n)|}$$

with $CGeom$ representing the geometric constraints of n , $CZones$ the zoning constraints, and $CShape$ the shape constraints.

These constraints differ in their evaluation, due to their different nature:

- Geometric constraints (such as parallelism) are strict, *i.e.* $CGeom \in \{0,1\}$;
- Zoning and shape constraints are fuzzy, *i.e.* $CZones, CShape \in [0, 1]$.

This is where the interaction between the learner module and the recognition engine lies. Although it is possible to verify the geometric constraints independently from the CD-CMG and Evolve duo (*e.g.* by defining the predicates directly in the domain module), the learner module has to query the recognition engine to check the satisfaction of the zoning (fuzzy positioning in CD-CMG) and shape (Evolve recognition score for a particular class) constraints.

Once a new element is matched with a knowledge graph node, the process of generating corrective feedback to the student is straightforward. The unsatisfied constraints of the node are translated into textual corrective feedback. Details on the typology of this feedback are developed in the next section.

5 Qualitative evaluation

In this section, we provide an overview of the feedback typology that we have defined for the first version of IntuiSketch. Feedback plays a crucial role in guiding the student during the exercise resolution mode. We take the example of the

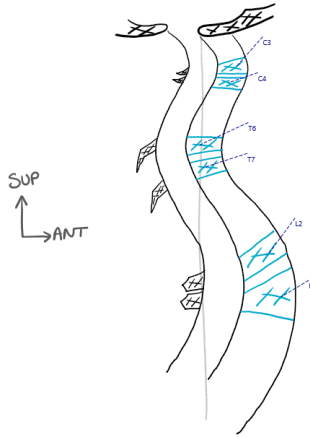


Fig. 9: Example of the spine exercise proposed by the teacher in author mode

spine to give a more detailed demonstration of the steps involved in producing this anatomy sketch from the beginning to the end. Fig. 9 shows the example provided by the teacher, which we intend to follow closely for this exercise.

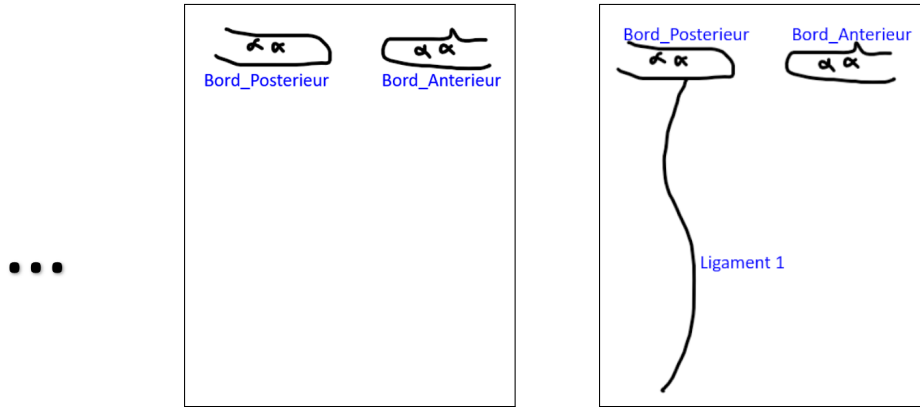
Fig. 10 illustrates steps in a construction problem we are currently working on. The segmented part of the sketch represents a specific element that the student needs to draw correctly. During this process, the system generates personalized feedback to help the student to improve their sketch (see captions for each sub-figure in Fig. 10).

As shown in the Fig. 10a, the IntuiSketch system interprets the student's drawings stroke by stroke. Once a stroke is recognized, the system displays the label of the recognized stroke.

Fig. 10b illustrates an example of positive feedback provided to the student upon successful drawing of the left ligament. The feedback displayed includes relevant information about the exercise and textual feedback, recognizing the student's correct representation.

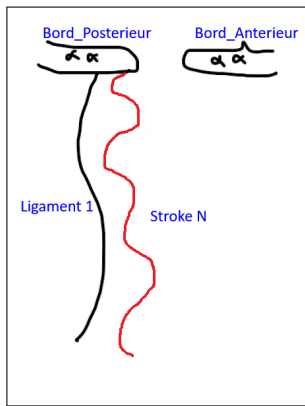
In contrast, Fig. 10c presents corrective feedback for a wrong drawing. Here, the student draws a stroke (strokeN), which the recognition system rejects because it doesn't match any CD-CMG production rule (the classification score is below a defined threshold). However, the shape of the stroke is similar to the shape of a ligament, leading the system to provide specific feedback on the shape and the geometric relationship between strokeN and ligament1 (i.e. they should be parallel). Fig. 10d highlights another case where the student draws a stroke that matches the cervical node. However, the positioning of the stroke is incorrect, as it should be in the top-left zone of the spine. Consequently, the system generates corrective feedback targeting both the zoning and shape constraints.

Fig. 10e represents the final step of the spine exercise. The system recognized and interpreted all sketches after having guided the student through all the steps,

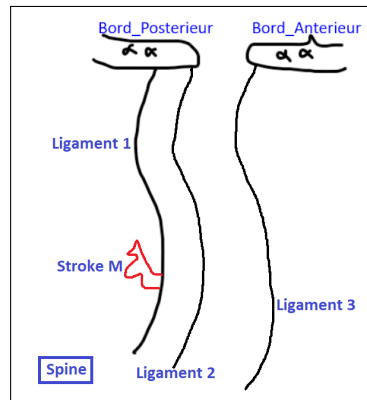


(a) Label displayed when the stroke is recognized

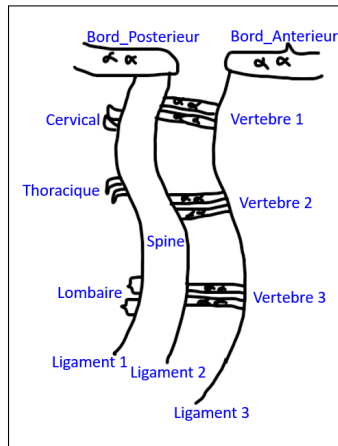
(b) Positive feedback on drawn ligament:
- The resolution of ligament 1 is complete.



(c) Corrective feedback on drawn ligament (in red):
- Stroke N and Ligament 1 had to be parallel.
- The shape of stroke N had to be Ligament.



(d) Corrective feedback on drawn cervical (in red):
- Stroke M had to be in the zone TopLeft.
-The shape of stroke M had to be Cervical.



(e) The complete example of the spine drawn step by step

Fig. 10: Personalized feedbacks generated to the student during his realization for the spine example

providing different types of feedback by comparing what the student had done to the model drawn by the teacher in Fig.9.

This example shows all the feedback we can generate to date, and, that work is ongoing.

6 Conclusion and perspectives

In this paper, we introduced IntuiSketch, an intelligent tutoring system for anatomy courses. The system combines online recognition techniques with tutoring techniques to provide personalized feedback to students. Online recognition is based on the CD-CMG grammar combined with the Evolve incremental classifier and the fuzzy positioning, which enables sketches to be interpreted in real time. The tutoring aspect is based on constraint modeling, which enables the analysis of the student's production in relation to the problem constraints and the generation of feedback.

A new tutoring architecture has been introduced for the IntuiSketch system, with an emphasis on the interaction between the recognition and tutoring aspects, and the combination of different kind of information -shape, position, geometric constraints- to model the domain knowledge.

Future work will focus on improving and extending the capabilities of IntuiSketch. We aim to infer composition rules dynamically based on reference sketches created by the teacher in author mode, thus automatically enriching the domain module. The objective is to evolve the grammar incrementally, adding new production rules each time the teacher introduces new concepts by drawing reference sketches. Furthermore, we will carry out more in-depth evaluations of the IntuiSketch system in real-life educational settings. This will include larger-scale deployments and the gathering of feedback from both teachers and students to assess the system's ability to improve learning experience using pen-based tablets.

Acknowledgements This work is funded by the ANR project number 21-CE38-0009. We would like to thank our collaborators from the LP3C laboratory (Psychology, Cognition, Behavior and Communication).

References

1. Alkhatlan, A., Kalita, J.: Intelligent tutoring systems: A comprehensive historical survey with recent developments. *International Journal of Computer Applications* **181**(43), 1–20 (Mar 2019). <https://doi.org/10.5120/ijca2019918451>, <http://www.ijcaonline.org/archives/volume181/number43/30402-2019918451>
2. Almaksour, A., Anquetil, E.: Iclass: Error-driven antecedent learning for evolving takagi-sugeno classification systems. *Applied Soft Computing* **19**, 419–429 (2014). <https://doi.org/https://doi.org/10.1016/j.asoc.2013.10.007>, <https://www.sciencedirect.com/science/article/pii/S1568494613003414>

3. Atilola, O., Valentine, S., Kim, H.H., Turner, D., McTigue, E., Hammond, T., Linsey, J.: Mechanix: A natural sketch interface tool for teaching truss analysis and free-body diagrams. *AI EDAM* **28**(2), 169–192 (2014). <https://doi.org/10.1017/S0890060414000079>
4. Carbune, V., Gonnet, P., Deselaers, T., Rowley, H., Daryin, A., Calvo, M., Wang, L.L., Keysers, D., Feuz, S., Gervais, P.: Fast multi-language lstm-based online handwriting recognition. *International Journal on Document Analysis and Recognition (IJ DAR)* p. 89–102 (2020). <https://doi.org/10.1007/s10032-020-00350-4>
5. Delaye, A., Anquetil, E., Macé, S.: Explicit fuzzy modeling of shapes and positioning for handwritten chinese character recognition. In: 2009 10th International Conference on Document Analysis and Recognition. pp. 1121–1125 (2009). <https://doi.org/10.1109/ICDAR.2009.141>
6. Fiorella, L., Mayer, R.E.: *Learning as a Generative Activity: Eight Learning Strategies that Promote Understanding*. Cambridge University Press (2015). <https://doi.org/10.1017/CBO9781107707085>
7. Krichen, O., Anquetil, E., Girard, N.: IntuiGeo: Interactive tutor for online geometry problems resolution on pen-based tablets. In: European Conference on Artificial Intelligence (ECAI) 2020. pp. 1842 – 1849. Santiago de compostela, Spain (Aug 2020), <https://hal.science/hal-02544384>
8. Kulik, J.A., Kulik, C.L.C.: Timing of feedback and verbal learning. *Review of Educational Research* **58**(1), 79–97 (1988). <https://doi.org/10.3102/00346543058001079>
9. Macé, S., Anquetil, E.: Eager interpretation of on-line hand-drawn structured documents: The dali methodology. *Pattern Recognition* **42**(12), 3202–3214 (2009). <https://doi.org/https://doi.org/10.1016/j.patcog.2008.10.018>, <https://www.sciencedirect.com/science/article/pii/S0031320308004482>, new *Frontiers in Handwriting Recognition*
10. Sarvadevabhatla, R.K., Kundu, J., R, V.B.: Enabling my robot to play pictionary: Recurrent neural networks for sketch recognition. In: *Proceedings of the 24th ACM International Conference on Multimedia*. p. 247–251. MM '16, Association for Computing Machinery, New York, NY, USA (2016). <https://doi.org/10.1145/2964284.2967220>, <https://doi.org/10.1145/2964284.2967220>
11. Wang, X., Chen, X., Zha, Z.: Sketchpointnet: A compact network for robust sketch recognition. In: 2018 25th IEEE International Conference on Image Processing (ICIP). pp. 2994–2998 (2018). <https://doi.org/10.1109/ICIP.2018.8451288>
12. Zhang, X., Huang, Y., Zou, Q., Pei, Y., Zhang, R., Wang, S.: A hybrid convolutional neural network for sketch recognition. *Pattern Recognition Letters* **130**, 73–82 (2020). <https://doi.org/https://doi.org/10.1016/j.patrec.2019.01.006>, <https://www.sciencedirect.com/science/article/pii/S0167865519300078>, *image/Video Understanding and Analysis (IUVA)*
13. Álvaro, F., Sánchez, J.A., Benedí, J.M.: An integrated grammar-based approach for mathematical expression recognition. *Pattern Recognition* **51**, 135–147 (2016). <https://doi.org/https://doi.org/10.1016/j.patcog.2015.09.013>, <https://www.sciencedirect.com/science/article/pii/S0031320315003441>