Computer Representation of Venn and Euler Diagrams

Diunuge B. Wijesinghe, Surangika Ranathunga, Gihan Dias

Department of Computer Science and Engineering, University of Moratuwa, Sri Lanka

{diunuge.10, surangika, gihan}@cse.mrt.ac.lk

***Abstract*—Venn & Euler diagrams are well-defined mathematical diagram types, which are the major representation methods of Set Theory. Although understanding of different diagram types such as charts and coordinate graphs has been addressed, no research has been done for Venn and Euler diagram interpretation from an image. Venn and Euler Diagrams exist in various media types such as printed format in books, raster images and vector images in electronic media. In this research, a methodology for set details extraction from a vector image is presented and Venn data representation is introduced, which can store Venn details extracted from a Venn or Euler diagram.**

***Keywords*— Diagram Understanding, Venn Diagram, Euler Diagram, Set Theory, Vector Images**

I. INTRODUCTION

Diagrams are a very important communication medium. This is especially the case with Mathematical diagrams. Although humans can easily interpret these diagrams, mathematical diagram understanding is a complex challenge for researchers. Diagram understanding is an important pre-requisite of various fields such as image database systems, and educational diagram grading systems. Significant research has been done to understand mathematical diagrams in few domains such as coordinate graphs [1,2] as well as charts [5] (bar charts, pie charts). However, there is no significant research done to interpret Venn and Euler diagrams.

We address the problem of computer understanding of Venn and Euler diagrams that are available in vector format by translating an image into an abstract data structure using domain knowledge of Venn and Euler diagrams. Venn Diagrams have been developed by John Venn (1843 - 1923) and became a common representation tool in propositional logic and related branches of Mathematics such as Boolean Algebra [7]. Venn diagrams can be described as diagrams that represent pictorial relations among sets. Venn diagrams are a specialized instance of a more general notation for representing relationships among a set of classes of concepts referred to as Euler diagrams developed by Leonard Euler [20].

Definition of Venn and Euler diagrams varies throughout the literature [20]. Generally, Venn and Euler diagrams can be defined as a finite set of labelled, closed curves. The closed curves in the diagram partition the plane into minimal regions, where each minimal region is a connected component of the plane inside a set of curves [20]. While a Venn diagram contains all the minimal regions, Euler diagrams omit the empty minimal regions. Normally Venn and Euler diagrams with two to three sets are being used in most of the educational contexts such as mathematics and science. Representation up to six sets is less complex in the graphical representation. Fig. 1 shows a Venn diagram and the corresponding Euler diagram of two sets for which the set of $A \cap \sim B$ is empty.

To address the problem of computer understanding of Venn and Euler diagrams, we present a methodology to extract set details from a vector image and produce the output as an object structure. Our system accepts input images as vector images in SVG (scalable vector graphics) format. Fig. 1 shows part of an SVG image.

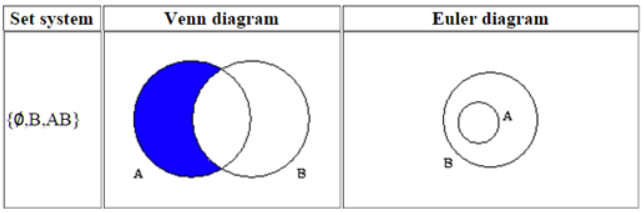

Fig. 1 Example of a Venn and an Euler Diagram

In vector images, sets are identified by labelled Jordan Curves, which are non-self-intersecting continuous closed paths in the plane [20]. In SVG, sets can be represented with circles, ellipses, rectangles and closed curves. In this research we only consider circles, ellipses and rectangles. However, same concepts can be extended for the arbitrary closed curves as well, though it is not currently handled. A set usually has an attached label. In the label identification, only nominal text labels are accepted as labels. Since the label identification is subjected to various ambiguity problems, various heuristics had to be used to identify correct labels.

After identifying the sets, minimal regions are identified. Then remaining text is classified as zone elements. After extracting all the Venn information, extracted data is represented by an object structure that can describe any Venn diagram. It contains set details and the zone details. Arrangement difference in the vector image of the Venn diagram can produce several outputs that are equivalent.

```
<svg width="580" height="400" xmlns="http://www.w3.org/2000/svg">
 <g>
  <title>Layer 1</title>
  <ellipse ry="97.5" rx="149" id="svg_1" cy="154" cx="197.5" stroke-width="1.5" stroke="#000" fill="#E6C36A"/>
  <ellipse ry="99" rx="160" id="svg_2" cy="151" cx="367.5" stroke-width="1.5" stroke="#000" fill="none"/>
  <ellipse ry="109.5" rx="144.5" id="svg_3" cy="256" cx="280" stroke-width="1.5" stroke="#000" fill="none"/>
  <rect stroke="#000" id="svg_14" height="348.999976" width="521.999994" y="30.50001" x="27.500014"
    stroke-opacity="null" stroke-width="1.5" fill="none"/>
  <text font-family="Helvetica, Arial, sans-serif" font-size="24" id="svg_4" y="75.5" x="96.5" fill-opacity="null"
    stroke-opacity="null" stroke-width="0" stroke="#000" fill="#000000">A</text>
  ...
 </g>
</svg>
```

Fig. 2 SVG XML structure of a Venn diagram

This method has been tested against a set of Venn and Euler diagrams produced by university students and secondary school students. We gave a question paper with four Venn and Euler a diagram questions taken from the Mathematics paper of the GCE O/L examination in Sri Lanka to university undergraduate and postgraduate students and collected answers in hand written format. Also, we have collected student answers from a secondary school examination that has Venn and Euler diagram questions. Collected answer sheets were converted to electronic format (SVG) before parsing. The diagrams were parsed using the system. Parsed output was manually checked against the original diagram for the validation of the system, which showed an accuracy of 89.61%.

This paper is arranged into following sections: Section II describes the related work. Abstract solution details of the complete research are included in Section III. Section IV provides an evaluation of the system. Finally, Section V concludes the paper with possible future extensions.

II. Related Work

In this section, we explore existing research related to mathematical diagram recognition, interpretation, diagram similarity measurements, and diagram data representation methods.

Mathematical diagram recognition, understanding, and evaluation is a relatively new research field. As an early attempt, Futrelle et al [1] presented a diagram understanding system to interpret diagrams based on constraint grammars. The system is capable of handling x-y graphs and gene diagrams in Biological domain [2]. Tsintsifas et al [8] developed a java based framework called DATsys that can be used to understand and evaluate diagrams. They were able to develop the diagram input system that can be scalable to various diagram types. This system was developed as an extension to existing Ceilidh Computer Assessment System [17]. Thomas et al [9, 10, and 12] developed a computer aided assessment system that can handle graph based diagrams such as Entity-Relationship diagrams and flow charts, where information can be represented as data nodes and relations between data nodes. Diagrams are interpreted using a basic set of units called “Minimal Meaningful Units”. Tsintsifas et al [8] developed a method to assess diagrams in formative assessment in which primary aim was to assist the process of learning. They followed the work of Brett et al [15] and developed a feedback system that can work with graph based diagrams. They also discussed a simple evaluation method and developed a grammar for E-R diagrams. Batmaz et al [18] developed a diagram drawing tool that can be used for semi-automated database diagram assessment. Huang et al [4, 5] developed a system that can understand chart images. They were able to recognize and identify various types of chart images (both 2D and 3D) and interpret those images and produce an XML output that can be used for further processing. Research from Futrelle et al [1] and Huang et al [4] worked with raw pixel images that were extracted from printed documents. They converted those pixel images to Vector Graphics format for further processing. Anderson et al [16] discussed the fundamental components of a diagrammatic processing system, (1) Means to input diagrams that can be a vision component or direct link to a diagram source, (2) Diagram representation to internally represent diagrams, (3) Storage management component and (4) Processing component that synthesizes and abstracts new knowledge from combinations of diagrammatic and other forms of knowledge representations. In early research, they dealt with many types of input system types based on their research focus.

None of above research addresses the issue of Venn or Euler diagram interpretation and we are not aware of any other general Venn or Euler diagram interpretation research work.

Embedded text in an image is a key factor to recognize and interpret image information. Since labelling has more freedom, text association is a hard problem. When using basic image input formats, pixel based or vector based, input text association problem arises, which leads to ambiguity problems in image understanding [3].

Futrelle et al [2] discussed the object association problem using special techniques in general by dividing the Cartesian space of the diagram into regions and use cell indexes of graphic primitives to improve the processing efficiency. Huang et al [5] developed a machine learning technique based on the decision graph technique to pre-process text to help text association. However, such machine learning techniques require a large data set for better accuracy.

Some diagram based assessment systems have added restrictions to the label choice in order to minimize the association problem. Diagram drawing tool by Batmaz et al [18] provides a set of labels that can be selected as inputs. DATsys [8] and OpenMark [9] systems provide label insertion areas but the user can decide the label he wants to enter. Even though such restrictions in input reduce the text association complexity, such restrictions are not

recommended in some situations such as examination testing systems.

III. Venn and Euler Diagram Parser

In this section we describe the design details of the Venn and Euler diagram parser.

A. High-Level Architecture

Fig. 3 shows the high-level architecture of the implemented system. Input diagram is given in the SVG format. Since it contains presentation and SVG specific data, SVG image has to be parsed and primitive shape details should be extracted (Fig. 2 shows part of an SVG image which represents a Venn diagram). These details are needed to build Venn and Euler data information.

From the parsed SVG, sets are identified by finding the Jordan curves. In this case, only circles, rectangles, and ellipses are considered. After finding the sets, relevant set labels are identified using a heuristic algorithm ("Set Label Identification" section). Only nominal labels are considered as valid set labels. Since text labels can be categorized into few types such as title, set labels and set elements, few heuristic parameters such as closeness to the set area boundary, nominal/ numeric labels, font size and relative position to set areas are considered to classify the set labels. If sets are labelled using arrows, relevant arrows are identified and the associated text with the arrow is considered as the set label. Generated temporary label is given to the set if there is no associated set label.

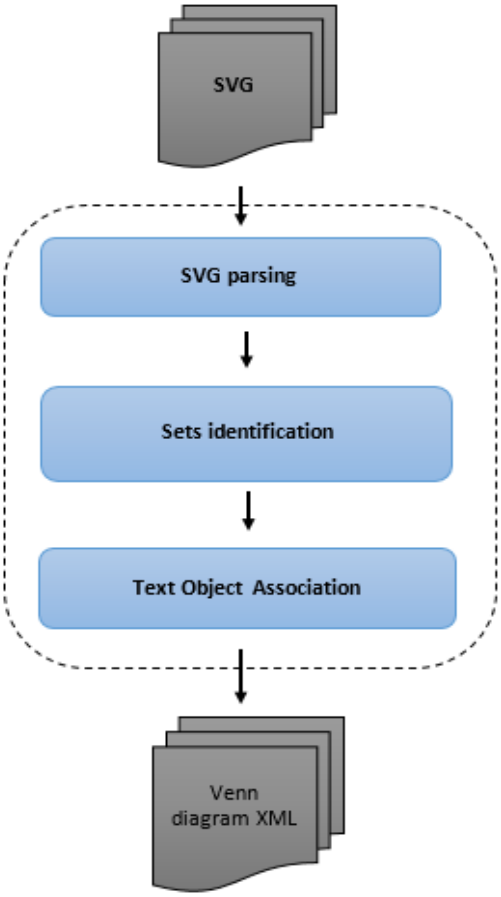

Fig. 3 High Level Architecture of the tool

At this stage, all the minimal regions are identified and minimal region geometric properties such as centroid of the minimal region area and size of the area are calculated for text association building.

Huang et al [4, 5] introduced text association technique using machine learning for charts. Since we have limited diagram database, we used a heuristic approach for the text association instead of a classification approach. Then the text labels are associated with the correct minimal regions or zones (set of minimal regions) based on the Venn diagram domain knowledge. Several heuristic algorithms are used to deal with human errors ("Text Association Mapping" section and followed the text label ambiguity resolution methods discussed by Futrelle et al [3]. Various SVG editors can produce different diagrams such as diagrams with different scale of objects, different font sizes and font types. These heuristic parameters depend on the source of the Venn and Euler diagrams.

After associating the text labels, Venn diagram is built using the extracted knowledge and output is created as an object structure, which contains only the Venn and Euler details without the initial Venn diagram presentation details such as orientations, and set curve shapes. This object structure can be easily represented using an XML schema. XML schema is influenced from the schema developed for charts by Huang et al [5].

B. SVG Parsing

SVG format is the standard W3C standard of vector graphics. SVG format is capable of handling Venn diagrams. Fig. 4 shows a Venn diagram drawn in SVG format.

SVG of a Venn diagram contains basic mandatory SVG features and optional details such as size and file type, grouping details, image titles, primitive object (rectangle, line, circle, etc.) shape details, primitive object presentation details (fill, stroke details, etc.), and text label details. In SVG parsing, only details required to generate Venn information are retained. Parsed SVG contains mostly geometrical details such as size, primitive object geometry details and text label details.

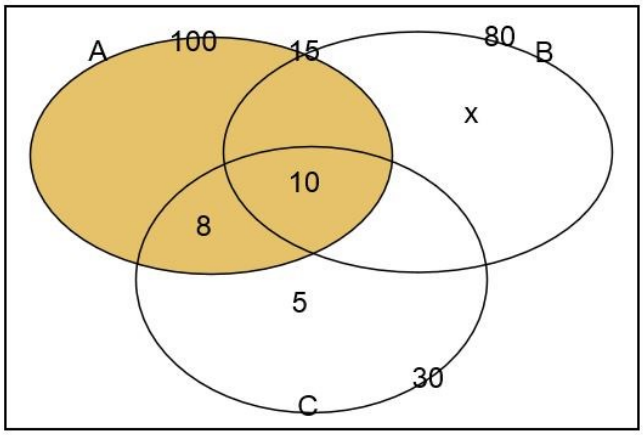

Fig. 4 Venn diagrams as an SVG image

C. Set identification

In Venn and Euler diagrams, sets are represented with Jordan curves (non-self-intersecting closed curves). In SVG diagrams, there are several possible primitive objects such as

circles, rectangles, ellipses and closed paths that can be considered as Jordan curves. We identified those closed curves in the SVG diagram from the SVG objects. In this research, only circles, rectangles, and ellipses are considered since the majority of the Venn and Euler diagrams are drawn using those shapes [20]. However, the same methodology can be extended to other primitive objects that act as Jordan curves.

D. Set Label Identification

After identifying sets, associated set labels have to be identified. Sets can be labelled in two ways; with arrows, and without arrows by putting the text label near the boundary of a set area. If labelled with arrows, arrow ending has to exist near the set area boundary. If an associated arrow is found for any set, then the associated text near the arrow tail is considered as the set label.

If there is no associated arrow for a set, then the closest nominal label near the boundary of the set area is considered as the set label. Fig 5 shows the boundary condition check for an ellipse. In the ellipse boundary condition check, only the centre (C) coordinates and horizontal radius (a) and vertical radius (b) are given in the SVG. Using these details, focal points (F_1, F_2) and the sum of distance to any point on the ellipse from focal points are calculated (Eq. 1, 2 and 3) from mathematical properties of an ellipse Fig. 6 shows the required mathematical properties.

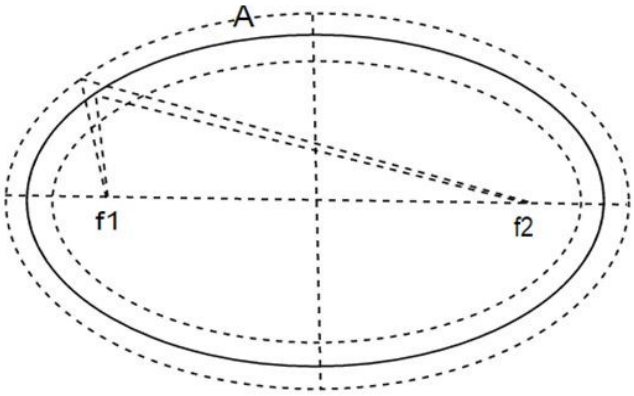

Fig. 5 Boundary condition check for an ellipse

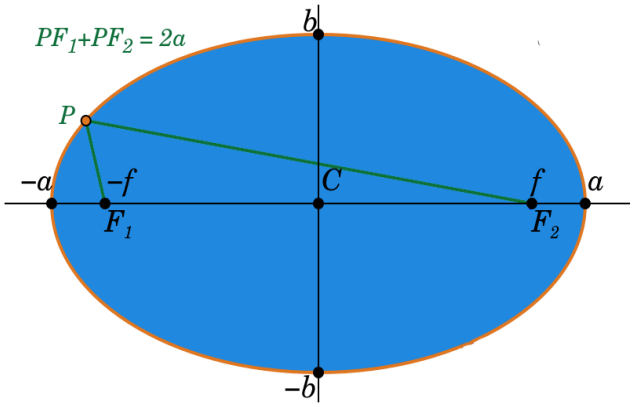

Fig. 6 Mathematical properties of an ellipse

$$f = sqrt(a^2 - b^2)$$ --- Eq. (1)

$$F1 = (Xc - f, Yc)$$ --- Eq. (2)

$$F2 = (Xc + f, Yc)$$ --- Eq. (3)

These are used to check the boundary conditions. The text labels appearing in the narrow area near a set boundary are considered as possible set labels for the considered set. From text labels that exist only in the considered set area boundary, the closest nominal label is selected as the label for the corresponding set. Closeness to the boundary of a set has an upper limit of T_1 * MINIMAL_FONT_SIZE. The minimum font size of an SVG image is extracted using all the text labels.

Arrows are presented as lines in SVG. If an arrow ending exists near a boundary of a set, closest nominal label (up to the upper limit of T_2 * MINIMAL_FONT_SIZE) of the other end of the arrow is considered as the set label. After set identification, sets without any label are given a generated label name. In the experiments, T_1 boundary parameter is selected as 1.5 and T_2 boundary parameter is selected as 5 based on the tuning phase results since closeness depends on the image scale that can be tuned.

E. Text Association Mapping

Algorithm 1 shows the algorithm of text label association. In a Venn or Euler diagram, elements associated with minimal regions are marked on the minimal region area. Since the minimal regions are constructed from combinations of set boundary parts, minimal regions normally have complex boundaries. Since the identification of minimal region boundaries is a complex task, minimal region centroid and the area of the minimal region are approximated by counting coordinates belonging to each region.

```
1. Calculate the centroid and the area of all minimal
       regions
2. Create pool of text labels except set labels
3. For (pool of text labels)
4. If (is text label on a border of a set)
5.       Text association with corresponding minimal
                region
6. Else
7.       Text association with relevant zone
8. Normalization
```

Algorithm 1. Pseudo code of the text association mapping

Then, the possible text that can be an element of a minimal region is filtered using the centroid and minimal region area. From the filtered text elements, correct text elements that are in the region are identified.

F. Venn Data Structure

Fig. 7 shows the XML schema corresponding to the Venn and Euler object structure. This structure is influenced by the XML structure introduced by Huang et al [5] for charts. In the Venn data XML structure, there are five top level tags; (1) type: Type of the diagram (Ex: - "Venn Diagram"), (2) Title: Diagram Title Label (If any), (3) no_of_sets: Number of sets in the diagram, (4) sets: set of sets, (5) data_set: minimal region and zone data.

Dataset order is ignored in the Venn information object structure. Depending on the set order, different Venn or

Euler diagrams may produce equivalent solutions. When reading the output XML ignoring the order of sets, it will produce the same Venn or Euler data.

For a given set, output XML can have a limited number (n!) of equivalent formats depending on the arrangement of sets. Variation of Venn or Euler diagram for 2 sets produces 2 equivalent XML formats. Figure 8 shows two equivalent formats for Venn diagrams with two sets given in Figure 9.

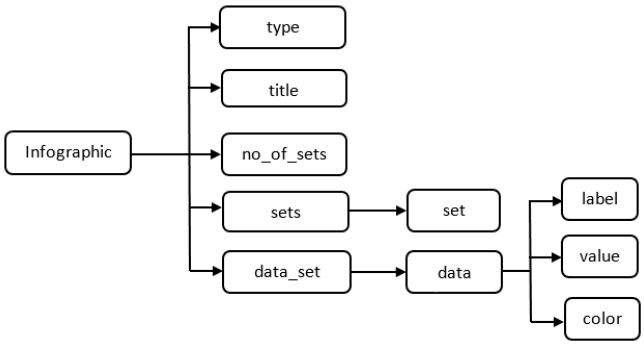

Fig. 7 Venn/ Euler data output XML structure

```
<infographic>
    <type>Venn Diagram</type>
    <title>Untitled</title>
    <no_of_sets>2</no_of_sets>
    <Sets>
        <set>A</set>
        <set>B</set>
    </Sets>
    <data_set>
        <zone><label>~A.~B</label><value>none</value>
            <color>none<color></zone>
        <zone><label>~A.B</label><value>18</value>
            <color>none<color></zone>
        <zone><label>A.~B</label><value>20</value>
            <color>none<color></zone>
        <zone><label>A.B</label><value>82</value>
            <color>none<color></zone>
        <zone><label>A</label><value>102</value>
            <color>none<color></zone>
        <zone><label>B</label><value>100</value>
            <color>none<color></zone>
    </data_set>
</infographic>

<infographic>
    <type>Venn Diagram</type>
    <title>Untitled</title>
    <no_of_sets>2</no_of_sets>
    <Sets>
        <set>B</set>
        <set>A</set>
    </Sets>
    <data_set>
        <zone><label>~B.~A</label><value>none</value>
            <color>none<color></zone>
        <zone><label>~B.A</label><value>20</value>
            <color>none<color></zone>
        <zone><label>B.~A</label><value>18</value>
            <color>none<color></zone>
        <zone><label>B.A</label><value>82</value>
            <color>none<color></zone>
        <zone><label>B</label><value>100</value>
            <color>none<color></zone>
        <zone><label>A</label><value>102</value>
            <color>none<color></zone>
    </data_set>
</infographic>
```

Fig. 8 XML representations for the Venn diagrams given in the Fig. 9

IV. EVALUATION

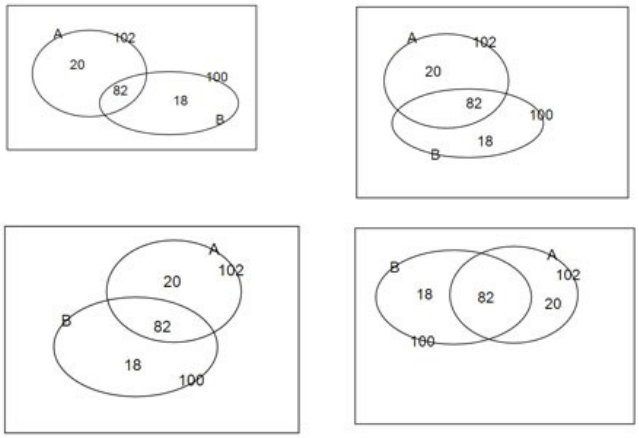

Fig. 9 Variation of Venn diagrams representing the same set information

Venn parser is tested and tuned using the Venn diagrams gathered from university undergraduates, which are drawn directly using an SVG editor. For further evaluation, we collected hand-written answer scripts that collectively contained 3 Venn and 4 Euler diagrams from university undergraduates and grade 10 school students. In total, we had 77 Venn and Euler diagrams. All the hand-drawn images were converted to vector format using an SVG editor before being given to the parser. Parser successfully parsed 69 of those diagrams correctly having a collective accuracy of 89.61% (Table I).

TABLE I. EVALUATION RESULTS

Diagram No.	Diagram Type	Correctly Parsed	Incorrectly Parsed	Accuracy
1	Venn	17	2	89.5%
2	Venn	12	0	100%
3	Euler	11	0	100%
4	Euler	11	0	100%
5	Euler	3	2	60.0%
6	Euler	5	3	62.5%
7	Venn	10	1	90%

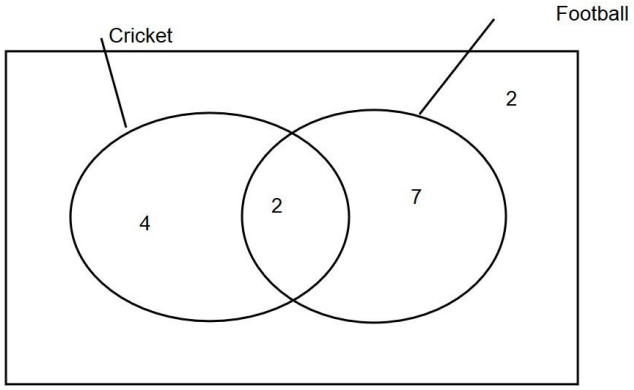

Fig. 10 Venn diagram that have ambiguous text labels

Diagram 1 and 2 contain two sets and remaining diagrams contains 3 sets. Drawings for diagram 1 were collected from school student answer scripts and remainder were collected from university undergraduates. Diagram 5 and 6 had fewer answers because of the higher complexity of the question. All of the parsing errors are due to the ambiguity of the text labels and some text labels being too far away from the arrows that are supposed to be arrow labels. Fig 10 shows Venn diagram with ambiguous set labels. The label "Cricket" can be recognized as set label of the universal set or set label represented by the first ellipse since it is close to the nearest line-end and the rectangle boundary. The set label "Football" may or may not be the set label for the set represented by the second ellipse, but it is too far away from the line-end. To resolve these text label ambiguities, a higher level of analysis is needed.

V. Conclusions

G. Future Work

Number of possible further extensions can be identified with respect to the implemented system.

In this research, sets can be drawn from few shapes such as rectangles, circles, and ellipses. This method can be extended to apply for sets drawn with any type of Jordan curves. In some Venn and Euler representations, one set label can have more than one Jordan curve. This method can be extended to address those representations.

Machine learning approaches can be used to improve the label classification and clustering. Currently, machine learning methods are not used since we do not have a significantly large vector image database.

This method can be easily extended into other vector formats of images since conversion methods of pixel images format into vector format are already developed. Also, this method can be extended to parse printed images.

H. Concluding Remarks

Diagram understanding is a complex problem in the computer research field. Structured diagrams can be dealt with using domain ontology, but the unstructured drawing understanding is a very complex problem. In particular, dealing with human drawn diagrams is difficult due to the ambiguity problems and human errors.

In this research, we have successfully established the required methods to interpret Venn and Euler diagrams represented in SVG vector format, and introduced a generic format to represent a Venn and Euler diagram. We believe that solutions we have introduced will help to develop systems related to image understanding such as automatic grading systems and image database systems.

Acknowledgment

This research is funded by the 2015 University of Moratuwa Senate Research Grant (SRC), and DL4D 2016 research grant.

References

[1]. R. P. Futrelle, I. Kakadiaris, J. Alexander, C. M. Carriero, N. Nikolakis, and J. M. Futrelle, "Understanding diagrams in technical documents," IEEE Computer, 1992, Vol. 25(7), pp. 75-78.

[2]. R. P. Futrelle and N. Nikolakis, "Efficient analysis of complex diagrams using constraint-based parsing," in Document Analysis and Recognition, in Proceedings of the Third International Conference on IEEE, 1995, Vol. 2, pp. 782-790.

[3]. R. Futrelle, "Ambiguity in visual language theory and its role in diagram parsing", in Proceedings 1999 IEEE Symposium on Visual Languages, 1999.

[4]. W. Huang, C. Tan, and W. K. Leow, (2005, August). "Associating text and graphics for scientific chart understanding," in Document Analysis and Recognition, in Proceedings of the Eighth International Conference on IEEE, 2005, pp. 580-584.

[5]. W. Huang, C. Tan, "A System for Understanding Imaged Infographics and Its Applications," in Proceedings of the 2007 ACM symposium on Document engineering - DocEng '07, 2007.

[6]. C.Tselonis, J. Sargeant, and M. M. Wood, "Diagram matching for human-computer collaborative assessment," in Proceedings of the 9th CAA Conference, 2005.

[7]. R. F. Gunstone and R. T. White, (1986). "Assessing understanding by means of Venn diagrams," Science Education, Vol 70(2), pp. 151-158, 1986.

[8]. A. Tsintsifas, "A framework for the computer based assessment of diagram based coursework," in Computer Science and Information Technology, 2002.

[9]. N. Smith, P. Thomas, and K. Waugh, "Interpreting imprecise diagrams," Diagrammatic Representation and Inference. Springer Berlin Heidelberg, 2004. 239-241.

[10]. P.Thomas, K. Waugh, and N. Smith, "Experiments in the automatic marking of ER-diagrams," ACM SIGCSE Bulletin, Vol 37(3), pp. 158-162, 2005.

[11]. P. Thomas, K. Waugh, and N. Smith, "Generalised diagramming tools with automatic marking," Innovation in Teaching and Learning in Information and Computer Sciences Vol 10(1), pp. 22-34, 2011.

[12]. P. Thomas, K. Waugh, and N. Smith, "Using patterns in the automatic marking of ER-diagrams," ACM SIGCSE Bulletin. Vol. 38(3), 2006.

[13]. J. Ambikesh and M. Shepperd, "The problem of labels in E-assessment of diagrams," Journal on Educational Resources in Computing (JERIC), Vol 8(4), pp. 12, 2009.

[14]. K. Waugh, P. Thomas, and N. Smith, "Toward the automated assessment of entity-relationship diagrams," in Second Workshop of the Learning and Teaching Support Network - Information and Computer Science (TLAD), 2004.

[15]. B. Bligh and C. A. Higgins, "Formative computer based assessment in diagram based domains." ACM SIGCSE Bulletin, Vol 38(3), pp. 98-102, 2006

[16]. M. Anderson and R. McCartney, "Diagram processing: Computing with diagrams," Artificial Intelligence, Vol 145(1), pp. 181-226, 2003.

[17]. E. Foxley, C. Higgins, T. Hegazy, P. Symeonidis, and A. Tsintsifas. "The coursemaster cba system: Improvements over ceilidh," in The Computer-Assisted Assessment Conference, 2001.

[18]. F. Batmaz and C. J. Hinde, "A diagram drawing tool for semi–automatic assessment of conceptual database diagrams," in Proceedings of the 10th CAA Conference, 2006.

[19]. G. Hoggarth and M. Lockyer, "An automated student diagram assessment system," ACM SIGCSE Bulletin, Vol. 30(3), 1998.

[20]. P. A. Rodgers, "A survey of Euler diagrams," Journal of Visual Languages & Computing, vol 25(3), pp.134-155, 2014.

[21]. (2014) The Vector graphics [Online]. Available: https://en.wikipedia.org/wiki/Vector_graphics