A fast geometric defuzzication operator for large scale information retrieval

Coupland, S. , Croft, D. and Brown, S.

Original citation & hyperlink:
Coupland, S. , Croft, D. and Brown, S. (2014) 'A fast geometric defuzzication operator for large scale information retrieval' in 2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE) (pp: 1143 - 1149). IEEE
http://dx.doi.org/10.1109/FUZZ-IEEE.2014.6891581

ISBN 978-1-4799-2073-0
DOI 10.1109/FUZZ-IEEE.2014.6891581

A Fast Geometric Defuzzication Operator for Large Scale Information Retrieval

Simon Coupland, David Croft and Stephen Brown

***Abstract*—In this paper we explore the centroid defuzzication operation in the context of specific data retrieval application. We present a novel implication and centroid defuzzication approach based on geometric fuzzy sets and systems. It is demonstrated that this new approach requires fewer operations and results in a significant reduction in processing time in our application.**

I. Introduction

Many Gallery, Library, Archive and Museum (GLAM) institutions posses sizeable collections of heritage objects and in recent years considerable effort has been expended in digitising information related to these heritage resources. Digitisation projects are conducted for a variety of reasons, typically focused on conservation, research and collection accessibility [1], but the result is that tens of millions (at least) of collection items exist digitally [2].

There is growing consensus among museum professionals and users about the importance of data integration between different collections to allow cross searching and data clustering that extends beyond the limited powers of basic keyword searching ([3], [4], [5], [6]. "'The nature of humanities data (being fuzzy, small scale, heterogeneous, of varying quality, and transcribed by human researchers) as opposed to scientific datasets (large scale, homogenous, numeric, and generated or collected/sampled automatically), means that novel computational techniques need to be developed to analyse and process humanities data for large scale projects"' [7].

Although cross-collection searches were always possible even with non-digitised objects, the reality is that they were very time and resource intensive and had, therefore, to be limited to both scope and number. Digitised heritage collections offer the possibility of easier cross collection searching. This would allow humanities researchers to investigate a broader range of sources and to conduct their investigations faster. Although the potential for cross-collection searching within GLAM collections is considerable, the nature of the information to be searched makes this a challenging problem. Record information within GLAM collections, is often imprecise and of uncertain accuracy. Coping with these issues requires elements from multiple domains, including but not limited to, Short Text Semantic Similarity (STSS), approximate string matching and fuzzy logic.

Simon Coupland is with the Centre for Computation Intelligence, David Croft and Stephen Brown are with the Knowledge Media and Design Group, De Montfort University, Leicester, LE1 9BH, United Kingdom (email: {simonc,dcroft, sbrown}@dmu.ac.uk).

This work was supported by Arts and Humanities Research Council under Project Code AH/J004367/1

Our previous research has demonstrated that semi-automated co-reference identification between GLAM collection records is possible despite the difficult nature of record information [8]. Having demonstrated its feasibility, our current focus is on implementing a system for ongoing and sustainable linkage of records in real heritage collections. At the present time we have records from fourteen different collections, amounting to more than 1.4 million.

The overall record similarity approach being used has been described in greater detail previously [8], although it has subsequently undergone further refinements. Of importance to this paper is that the final stage of the record pair comparison process is a Mamdani Fuzzy Inference System (FIS) which produces a defuzzified centroid. Whilst the sets and rules used by that FIS are very simple and have a low computational costs, the sheer number of comparisons to be made means that the cumulative time required for defuzzification is considerable.

Under our co-reference identification process, the total number of pair comparisons needed to compare r records is $r^2 - r$. Assuming that we wish to find co-reference candidates for every one of the ≈ 1.4 million records collected as part of this project so far, a total of 1.96×10^{12} comparisons would be required.

While the centroid was initially calculated using a discretisation approach in order to demonstrate that the rest of the co-reference approach was working as expected, this approach was (as expected) time consuming. Geometric defuzzification promises to be significantly cheaper computationally than discretisation techniques, but even with geometric approaches the number of comparisons represent a significant processing and therefore time cost. In order to process our large number of records is was therefore necessary to optimise every stage of processing as much as possible.

In this paper we describe the minimal computational cost geometric defuzzification process we use in order to produce dramatic processing throughput improvements for the overall co-reference identification process.

The reminder of the paper is structured as follows: Section II presents related work which underpins our new approach, Section III presents the novel geometric implication and defuzzification approach, Section IV presents a comparison between our approach and a discrete implementation and finally Section V concludes this work.

II. Related Work

The approach presented in this paper is a form of geometric fuzzy sets and systems approach first outlined in two papers by Coupland *et al* [9], [10]. Geometric fuzzy systems treat fuzzy sets (including type-2 interval and general) as geometric objects made up of simple geometric primitives. Coupland *et al* considered line segments as their geometric primitive when dealing with type-1 fuzzy sets. This required line segment intersection calculations [11] and a modified version of the Bentley-Ottman plane sweep algorithm [12] to compute logical operations on fuzzy sets. However, in this paper we use triangles as our geometric primitive and rely on the simple fact that a triangles area is given by half it's base times it's height and the centroid of a triangle is given by the arithmetic mean of it's three apexes. As with all geometric fuzzy system we restrict ourselves to only using Mamdani style rule based systems and only using minimum and maxmium for t-norms and t-conorms. We also only consider the centroid defuzzifier. The main reason for this is that we found it worked well in our application and therefore our motivation for this work was simply to improve computation time. There are a number of other approaches to defuzzification which are efficient (centre of sums, mean of maxima, height. See [13] for details), however we wished to maintain the behaviour of our prototype rule base so therefore stuck to the centroid.

III. Fast Defuzzification Algorithm

We now examine how to perform the geometric implication and centroid defuzzication for triangular membership functions. We do not cover all potential combinations, however we cover all those required for our application.

A. Centroid of a Single Symmetrical Triangle

Consider the fuzzy set whose membership function is a single symmetrical triangle depicted in Figure 1. The

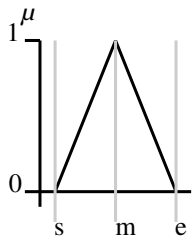

Fig. 1. A Single Symmetrical Triangular Fuzzy Set.

membership function of this set is simply a triangle, therefore it's area is half the base length multiplied by the height. The centroid can be calculated as the mean of the three co-ordinates which make up the triangle. The area and the centroid of this fuzzy set are given in equations 1 and 2.

$$A = \frac{e-s}{2} \tag{1}$$

$$C = \frac{e+m+s}{3} \tag{2}$$

where A is area and C is the centroid. For this triangle $e-m = s-m$, therefore the centroid is simply m. Now suppose this set is used as a consequent of a Mamdani style rule with a firing strength of μ. Such a situation is depicted in Figure 2(a). The shape of the resultant consequent set is a

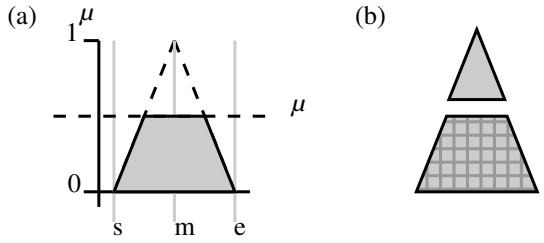

Fig. 2. A Single Symmetrical Triangular Consequent Fuzzy Set Under Firing Strength μ.

trapezoid which is shown as the grey shaded area in Figure 2(a). This trapezoid shape can be easily constructed from two triangles. If we take the triangle depicted in Figure 2(a) from the triangle in Figure 1 we arrive at the trapezoid which would result from the impaction of the rule firing strength μ on the triangular consequent set. The area of this set is given by subtraction of the triangle areas. The area of the smaller second triangle is given by equation 3.

$$A = \frac{(e-s)\times(1-\mu)}{2}\times(1-\mu) \tag{3}$$

Notice that the base of the second triangle is calculated by multiplying the base of the original triangle by $(1-\mu)$, with $(1-\mu)$ being a simple scalar. We can rewrite the area of the resultant trapezoid as:

$$A = \frac{(e-s)\times(1-\mu)^2}{2} \tag{4}$$

The centroid of the trapezoid is clearly given by:

$$C = \frac{e+m+s}{3} = m \tag{5}$$

B. Centroid of a Non-Contained Pair of Symmetrical Triangles

Of course, in a Mamdani system consequent fuzzy sets must also be combined with the logical AND before defuzzifcation. Therefore, the next situation we are concerned with is a pair of triangular fuzzy sets under firing strengths of μ_1 and μ_2 respectively. We begin by looking at a pair of symmetrical triangles where $s_1 \leq s_2$, $m_1 \leq m_2$ and $e_1 \leq e_2$ as depicted in Figure 3 which we will refer to as non-contained.

The particular pair of triangles in Figure 3 can be deconstructed into five separate triangles which can be used to calculate the area and centroid of the resultant set. For the sake of generality we must include a sixth triangle not immediately apparent in Figure 3. In Figure 3 both μ_1 and μ_2 are greater than the y-component of the point where the line segments (m_1, e_1), (s_2, m_2) intersect. This may not always be the case as demonstrated by the pair of triangles depicted in Figure 4.

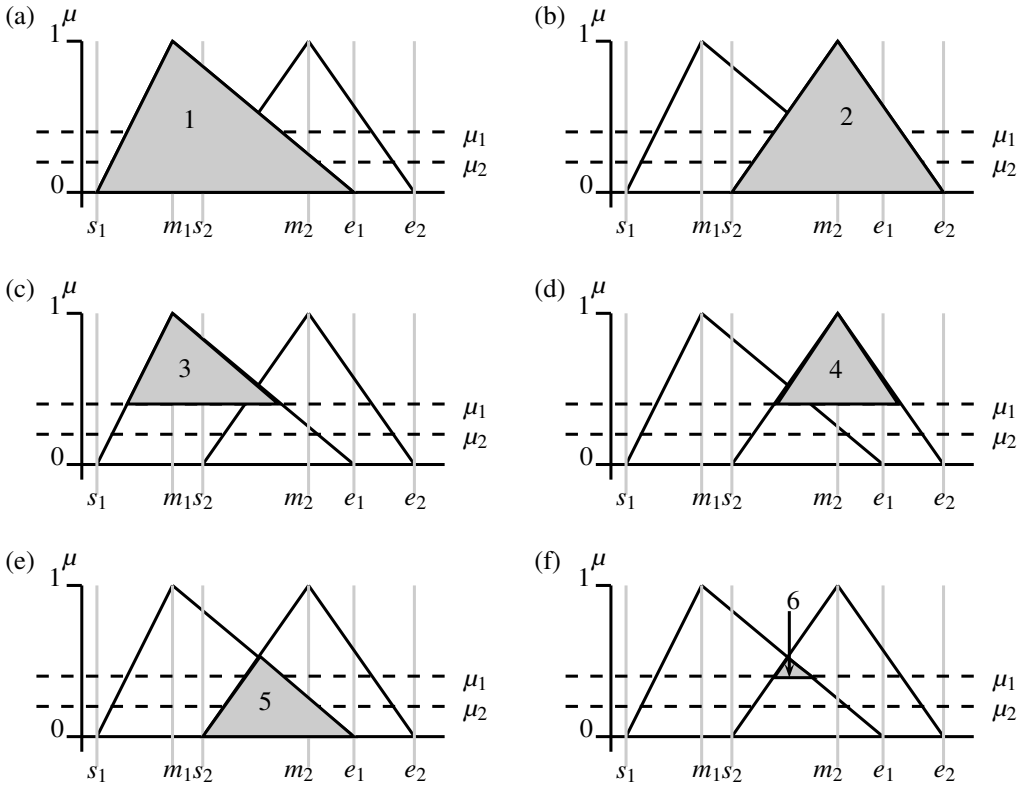

Fig. 5. A Second Pair of Symmetrical Triangular Fuzzy Sets.

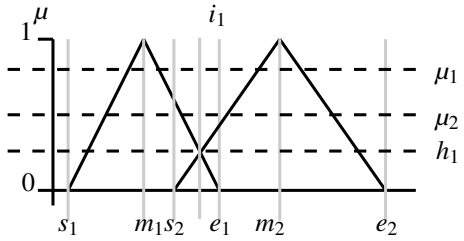

Fig. 3. A Pair of Symmetrical Triangular Fuzzy Sets.

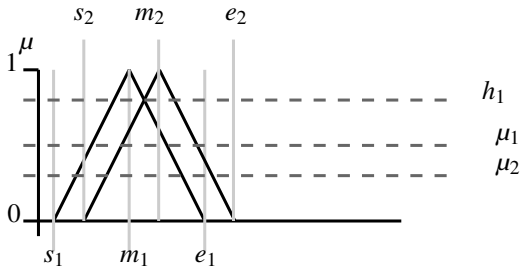

Fig. 4. A Second Pair of Symmetrical Triangular Fuzzy Sets.

A pair of symmetrical triangular fuzzy sets may be broken down into six triangles. The first two triangles are straightforward: s_1, m_1, e_1 (Figure 5 (a)) and s_2, m_2, e_2 (Figure 5 (b)). The second two triangles are used to form trapezoids formed by the application of the implication operator with the respective rule firing strengths of μ_1 (Figure 5 (c)) and μ_2 (Figure 5 (d)). The final pair of triangles we have to consider are formed by the intersection of the original two triangles. Clearly, the intersection of two triangles results in a triangle (Figure 5 (e)), however it may be that the implication operator also acts on this triangle resulting in a sixth and final triangle (Figure 5 (f)). The areas and centroids of each of these triangles are given in table I. When the situation depicted in Figure 3 occurs, the area of triangle 6 will be 0 and not contribute towards the centroid calculation.

Triangle	Area	Centroid
1	$\frac{(e_1-s_1)}{2}$	m_1
2	$\frac{(e_2-s_2)}{2}$	m_2
3	$\frac{(e_1-s_1)\times(1-\mu_1)^2}{2}$	m_1
4	$\frac{(e_2-s_2)\times(1-\mu_2)^2}{2}$	m_2
5	$\frac{(e_1-s_2)}{2}\times h_1$	i_1
6	$\frac{(e_1-s_2)\times(h_1-(h_1\wedge(\mu_1\vee\mu_2)))^2}{2}$	i_1

TABLE I
AREAS AND CENTROID OF THE TRIANGLES MAKING UP TWO NON-CONTAINED INTERSECTING SYMMETRICAL TRIANGLES.

The centroid C of the final consequent set is given by the

weighted average of each triangles area (equation 6).

$$C = \frac{C_1 \times A_1 + C_2 \times A_2 - C_3 \times A_3 - C_4 \times A_4 - C_5 \times A_5 + C_6 \times A_6}{A_1 + A_2 - A_3 - A_4 - A_5 + A_6} \tag{6}$$

However, we are concerned primarily with operational efficiency, our goal is to minimise computation time and we will modify equation 6 to make use of any precomputed values we can. Let α and β be precomputed values as follows.

$$\alpha = A_1 \times C_1 + A_2 \times C_2 - A_3 \times C_3 \tag{7}$$

$$\beta = A_1 + A_2 - A_3 \tag{8}$$

Note that all terms in equations 7 and 8 are known beforehand and are listed in Table I, therefore the centroid can be given by equation 9.

$$C = \frac{\alpha - C_4 \times A_4 + C_5 \times A_5 - C_6 \times A_6}{\beta - A_4 + A_5 - A_6} \tag{9}$$

C. The Centroid of A Fully Contained Pair of Symmetrical Triangles

We now move on to look at a pair of symmetrical triangles, which we refer to as fully contained, where $m_1 \leq s_2 \leq e_1$, $m_1 \leq m_2 \leq e_1$ and $m_1 \leq e_2 \leq e_1$ as depicted in Figure 6.

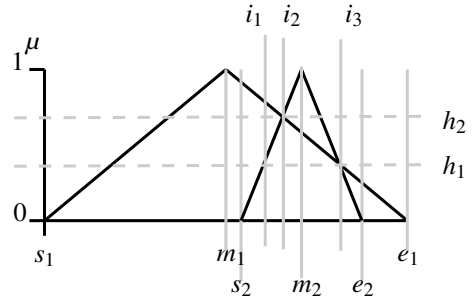

Fig. 6. A Pair of Fully Contained Symmetrical Triangular Fuzzy Sets.

The centroid of the conjunction of these two triangular fuzzy sets can be calculated from five triangles. Table II gives the area and centroid of each of these triangles and Figure 7 depicts the fourth and fifth triangles in Table II. The overall centroid is given by equation 10.

$$C = \frac{A_1 \times C_1 - A_2 \times C_2 + A_3 \times C_3 - A_4 \times C_4 - A_5 \times C_5}{A_1 - A_2 + A_3 - A_4 - A_5} \tag{10}$$

Equation 10 can be extended to include the implication operation We however, leave this to the reader.

D. Centroid of Four Non-Contained Symmetrical Fuzzy Sets

We move on to look at the exact problem faced in our data retrieval application, namely an efficient way of calculating the centroid of the output of four Mamdani rules. In our rule base the four fuzzy sets happen to be symmetrical and non-contained and this is the reason we have pursued the efficient defuzzication approach in the way described in this paper. Figure 8 depicts this situation where each of the four

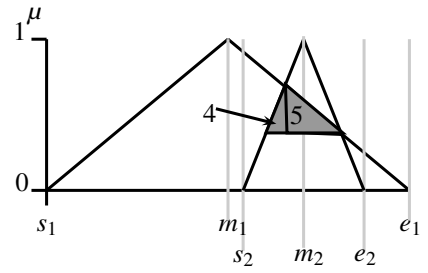

Fig. 7. A Pair of Fully Contained Symmetrical Triangular Fuzzy Sets.

Triangle	Area	Centroid
1	$\frac{(e_1 - s_1)}{2}$	m_1
2	$\frac{(e_2 - s_2)}{2}$	m_2
3	$\frac{(i_3 - i_1)}{2} \times (1 - h_1)$	m_2
4	$\frac{(i_2 - i_1)}{2} \times (h_2 - h_1)$	$\frac{i_1 + i_2 + i_2}{3}$
5	$\frac{(i_3 - i_2)}{2} \times (h_2 - h_1)$	$\frac{i_1 + i_2 + i_2}{3}$

TABLE II
AREAS AND CENTROID OF THE TRIANGLES MAKING UP TWO FUZZY CONTAINED INTERSECTING SYMMETRICAL TRIANGLES.

rules has some firing strength ($\mu_1 \ldots \mu_4$) which is implied across the respective consequents which are then combined and defuzzified. For this problem we must use 14 separate triangles to calculate the implication and defuzzification operation. These triangles follow from the triangles we used to calculate implication and centroid for a pair of non-contained symmetrical triangles. The areas and centroids of each of these 14 triangles are listed in Table III.

The only terms unknown before the implication and defuzzification are the rule firing strengths $\mu_1 \ldots \mu_4$. Any area or centroid listed in Table III not containing these terms can be computed ahead of time. This means all centroids are known ahead of time and areas $1 \ldots 7$ may be precomputed and may the combination of these first seven areas and centroids. In addition several components (i.e. $(e_1 - s_1)$) of the remaining areas may be precomputed. Let α and β be precomputed terms given by equations 11 an 12 respectively.

$$\alpha = \sum_{i=1}^{4} C_i \times A_i - \sum_{i=5}^{7} C_i \times A_i \tag{11}$$

$$\beta = \sum_{i=1}^{4} A_i - \sum_{i=5}^{7} A_i \tag{12}$$

The final centroid may then be given by equation 13. It is this equation which is used in the following section to achieve a reduction in the computation time of a large scale data retrieval application where all terms which can

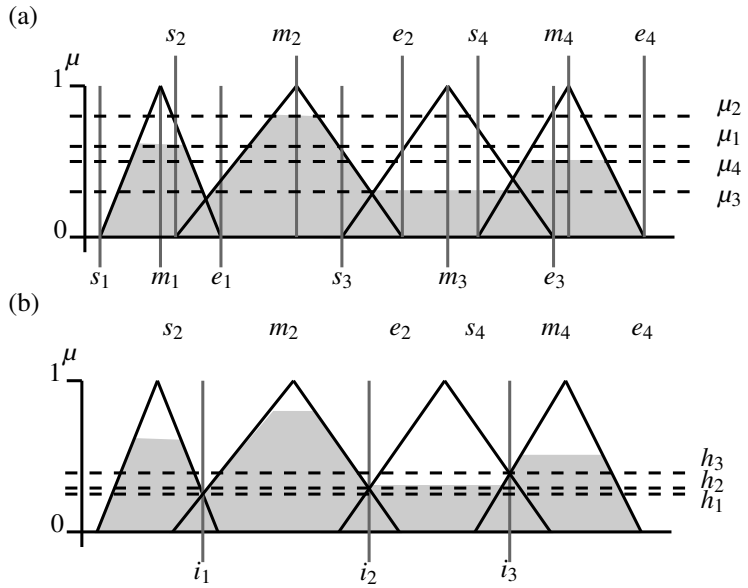

Fig. 8. Four Fully Contained Symmetrical Triangular Fuzzy Sets.

Triangle	Area	Centroid
1	$\frac{(e_1-s_1)}{2}$	m_1
2	$\frac{(e_2-s_2)}{2}$	m_2
3	$\frac{(e_3-s_3)}{2}$	m_3
4	$\frac{(e_4-s_4)}{2}$	m_4
5	$\frac{(e_1-s_2)}{2} \times h_1$	i_1
6	$\frac{(e_2-s_3)}{2} \times h_1$	i_2
7	$\frac{(e_3-s_4)}{2} \times h_1$	i_3
8	$\frac{(e_1-s_1)\times(1-\mu_1)^2}{2}$	m_1
9	$\frac{(e_2-s_2)\times(1-\mu_2)^2}{2}$	m_2
10	$\frac{(e_3-s_3)\times(1-\mu_3)^2}{2}$	m_3
11	$\frac{(e_4-s_4)\times(1-\mu_4)^2}{2}$	m_4
12	$\frac{(e_1-s_2)\times(h_1-(h_1\wedge(\mu_1\vee\mu_2)))^2}{2}$	i_1
13	$\frac{(e_2-s_3)\times(h_2-(h_2\wedge(\mu_2\vee\mu_3)))^2}{2}$	i_2
14	$\frac{(e_3-s_4)\times(h_3-(h_3\wedge(\mu_3\vee\mu_4)))^2}{2}$	i_3

TABLE III
AREAS AND CENTROID OF THE TRIANGLES MAKING UP FOUR INTERSECTING NON-CONTAINED SYMMETRICAL TRIANGLES.

	Number of Operations		
Operation	Novel Approach	201 Points	10 Points
$+$	14	402	22
$-$	7	0	0
$\times$	21	201	11
$/$	8	1	1
$\wedge$	3	804	44
$\vee$	3	603	33

TABLE IV
A COMPARISON OF THE COMPUTATIONAL COMPLEXITY OF OUR APPROACH WITH THE STANDARD APPROACH WITH TWO LEVELS OF DISCRETISATION.

be precomputed are precomputed.

$$C = \frac{\alpha - \sum_{i=8}^{11} A_i + \sum_{i=12}^{13} A_i}{\beta - \sum_{i=8}^{11} A_i + \sum_{i=12}^{14} A_i} \tag{13}$$

We now examine the computational complexity of our novel implication and defuzzification operation. We compare the number of operations required to compute the centroid using this new operation with the standard level of discretisation used in Matlab. Matlab by default will divide a consequent domain into 201 discrete points. For four Mamdani rules this will require the minimum of each point with the firing strength to be taken, the maximum of all these needs to be taken before the weighted sum of each pair is calculated. We also consider a much lower level of discretisation: 11 discrete points in the domains which we consider to be too coarse to be practical, but useful in the comparison of computational complexity. Table IV summarises the number of operations required for each approach in terms of $+$, $-$, $\times$, $/$, $\wedge$ and $\vee$. Clearly our novel approach has a much lower computational complexity

TABLE V
AVERAGE PROCESSING TIME REQUIRED.

Records	Time (seconds)	
	Discrete	Geometric
10	10.92	9.87
55	11.00	9.99
100	11.10	10.13
550	12.84	11.82
1000	15.99	14.16
5500	125.13	73.19
10000	375.56	196.97
55000	6368.61	2040.13

which we go on to demonstrate within an application in the following section. In terms of the accuracy of the results, as the approach taken is geometric the results are completely accurate [14], by definition more accurate than any discrete approach can be.

IV. TESTING

In order to measure the performance improvement offered by our geometric implementation, the overall co-reference identification system was run using both defuzzification approaches and the time required for it to finish processing was recorded in each case.

It was expected that the processing time required would increase exponentially as the number of records being processed increased. It was also expected that the geometric defuzzification approach would produce significantly faster processing times.

The system was run against groups of 10, 55, 100, 550, 1000, 5500 and 10000 records. The same records were used in testing both defuzzification methods and each test was run 10 times to produce a mean average for each set of records processed.

Testing was conducted on an Intel 3.10GHz quad core machine (i5-2400) with 8GB of RAM. As the co-reference identification software is multi-threaded, it was allowed to use all cores. When defuzzifying using a discrete approach, 201 points were used for each centroid calculated.

The average processing time results can be seen in table V. Based on our results we are able to make predictions for the time required to process full sets of records. Assuming 1.4 million records and using the discrete defuzzification approach, a total processing time of 3525517 seconds (40.8 days) is predicted. Using geometric defuzzification, that time is reduced to only 849318 seconds (9.8 days).

V. CONCLUSION

In this paper we have given a novel method for calculating the implication and centroid in a Mamdani system using triangular membership functions and minimum and maximum. We have not examined every case, however we have covered every circumstance required in out particular application, information matching and retrieval across museum collections. Our method requires less computation than a discrete approach resulting in a significant increase of processing speed in our application. In future work we will show how the approach can be used, how it can be generalised and what the performance implications of doing so are.

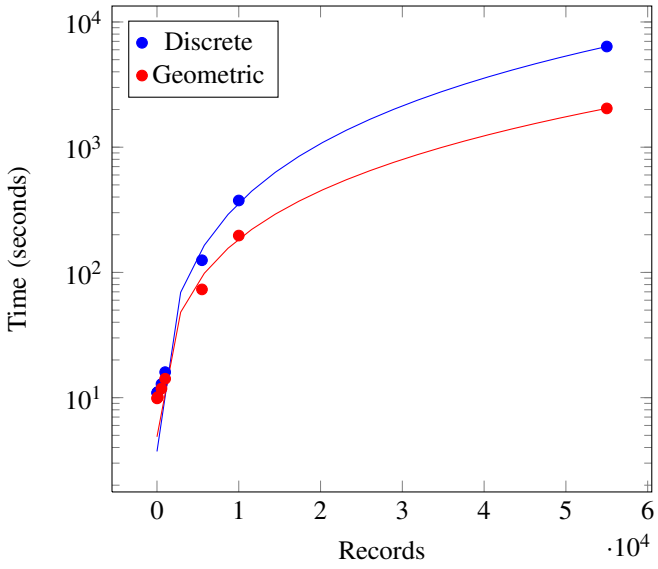

Fig. 9. Number of records processed versus time.

REFERENCES

[1] "Status of technology and digitization in the nation's museums and libraries," Washington, DC, 2006. [Online]. Available: http://web.archive.org/ web/ 20060926090433/ http://www.imls.gov/ resources/ TechDig05/ Technology%2BDigitization.pdf

[2] J. Purday, "Breaking new ground: Europeana annual report and accounts 2011," Europeana Foundation, Tech. Rep., June 2012. [Online]. Available: http://pro.europeana.eu/documents/858566/ade92d1f-e15e-4906-97db-16216f82c8a6

[3] B. Batjargal, T. Kuyama, F. Kimura, and A. Maeda, "Linked data driven multilingual access to diverse japanese ukiyo-e databases by generating links dynamically," *Literary and Linguistic Computing*, vol. 28, no. 4, pp. 522–530, 2013. [Online]. Available: http://llc.oxfordjournals.org/content/28/4/522.abstract

[4] A. de Polo, "Digital environment for cultural interfaces: Promoting heritage, education and research," in *Proceedings of Museums and the Web 2011*, April 2011.

[5] D. Henry and E. Brown, "Using an RDF data pipeline to implement cross-collection search | museumsandtheweb.com," in *Proceedings of Museums and the Web 2012*, Sep. 2011.

[6] T. Kamura, H. Takeda, I. Ohmukai, F. Kato, T. Takahashi, and H. Ueda, "Study support and integration of cultural information resources with linked data," in *Culture and Computing (Culture Computing), 2011 Second International Conference on*. IEEE, 2011, pp. 177–178.

[7] M. Terras, "The potential and problems in using high performance computing in the arts and humanities: The researching e-science analysis of census holdings (reach) project," *Digital Humanities Quarterly*, vol. 3, no. 4, 2009.

[8] D. Croft, S. Coupland, and S. Brown, "A hybrid approach to co-reference identification within museum collections," in *Computational Intelligence for Engineering Solutions (CIES), 2013 IEEE Symposium on*, 2013, pp. 110–117.

[9] S. Coupland and R. John, "Geometric Type-1 and Type-2 Fuzzy Logic Systems," *IEEE Transactions on Fuzzy Systems*, vol. 15, no. 1, pp. 3–15, February 2007.

[10] ——, "A Fast Geometric Method for Defuzzification of Type-2 Fuzzy Sets," *IEEE Transactions on Fuzzy Systems*, vol. 16, no. 4, pp. 929–941, 2008.

[11] P. Bourke, "Intersection point of two lines (2 dimensions)," April 1989, available at http://paulbourke.net/geometry/pointlineplane/.

[12] J. L. Bentley and T. A. Ottmann, “Algorithms for reporting and counting geometric intersections,” *IEEE Transactions on Computing*, vol. 28, no. 9, pp. 643–647, 1979.
[13] Timothy J. Ross, *Fuzzy Logic with Engineering Applications*, 3rd ed. Wiley, New Jersey, 2009, iSBN: 978-0-470-74376-8.
[14] S. Coupland and R. John, “On the Accuracy of Type-2 Fuzzy Sets,” in *Proc. FUZZ-IEEE 2007*, London, UK, July 2007, pp. 131 – 136.