Lappeenranta-Lahti University of Technology LUT

School of Engineering Science

Degree Programme in Computational Engineering and and Technical Physics

Intelligent computing

Markus Lindén

Fusion of Semi-Synthetic Retinal Image Segmentations

Bachelor's Thesis

Examiner: Professor Lasse Lensu

Supervisor: Professor Lasse Lensu

Abstract

Lappeenranta-Lahti University of Technology LUT
School of Engineering Science
Degree Programme in Computational Engineering and Technical Physics
Intelligent Computing

Markus Lindén

Fusion of Semi-Synthetic Retinal Image Segmentations

Bachelor's Thesis

2019

33 pages, 12 figures, 4 tables.

Examiner: Professor Lasse Lensu

Supervisor: Professor Lasse Lensu

Keywords: Expert annotation, ground truth, STAPLE, STAPLER, COLLATE, SIMPLE, majority vote, retinal image, image segmentation

The aim of this thesis was to get acquainted with and compare the performances of several label fusion algorithms experimentally using retinal image segmentations. Because the initial retinal image data could not be used alone as the input to the label fusion algorithm, several sets of synthetic segmentations were created. The data sets were created using BristolDB retinal image database that had segmentations for exudates in retinal images and the process was documented in detail. The best performing fusion algorithm was STAPLE, although there were no major differences in the performances of the used algorithms.

TIIVISTELMÄ

Lappeenrannan-Lahden teknillinen yliopisto LUT
School of Engineering Science
Laskennallisen tekniikan koulutusohjelma
Älykäs laskenta

Markus Lindén

Silmänpohjakuvien puolisynteettisten segmentointien yhdistäminen

Kandidaatintyö

2019

33 sivua, 12 kuvaa, 4 taulukkoa.

Ohjaaja: Professori Lasse Lensu

Tarkastaja: Professori Lasse Lensu

Hakusanat: Asiantuntijatieto, STAPLE, STAPLER, COLLATE, SIMPLE, enemmistöpäätös, silmänpohjakuva, kuvien segmentointi

Tämän kandidaatintyön tavoitteena oli tutustua ja vertailla useaa eri kuvasegmentointien fuusiointimenetelmää kokeellisesti käyttäen silmänpohjakuviin tehtyjä ryhmittelyjä. Koska alkuperäistä silmänpohjakuvadataa ei voitu yksinään käyttää kuvasegmentointien fuusiointimenetelmien syötteenä, tuotettiin useita puolisynteettisiä aineistoja kuvasegmentointien fuusiointimenetelmien vertailua varten. Nämä aineistot tuotettiin BristolDB silmänpohjakuvatietokantaa hyödyntäen, johon on merkitty silmänpohjakuvissa esiintyvät eksudaatit.
Parhaiten synteettisesti luodulla datalla toimiva kuvasegmentointien fuusiointimenetelmä oli STAPLE, vaikkakaan eri kuvasegmentointien fuusiointimenetelmien välillä ei ollut suurta eroa.

Contents

List of abbreviations

BristolDB	Bristol retinal image data set
COLLATE	Consensus level, labeler accuracy and truth estimation
EBM	Evidence-based medicine
EM	Expectation-maximization
MV	Majority vote
RGB	Red green blue
SIMPLE	Selective and iterative method for performance level estimation
STAPLE	Simultaneous truth and performance level estimation
STAPLER	Simultaneous Truth and Performance Level Estimation with Robust extensions

1 INTRODUCTION

1.1 Background

Evidence-based medicine (EBM) is the current practice used in many sub-fields of medical science. In EBM the medical professionals base their decisions on the patients' biomedical measurements and scientific knowledge [5]. Images are an important resource in EBM due to the versatile possibilities they offer in providing information about the patients' organs. Because of EBM medical, professionals can provide a diagnosis to the patients in a more complete and timely fashion [23].

Eye diseases have become one of the rapidly increasing health threats worldwide. For example, diabetes causes abnormalities in the retina (diabetic retinopathy), kidneys (diabetic nephropathy), and nervous system (diabetic neuropathy). The diabetic retinopathy and other eye-related diseases are diagnosed from eye fundus images by medical experts who look for special lesions in the images [23].

Having manually segmented retinal images by a single expert is of major benefit for the purpose of researching ways to automate the detection of lesions, but having multiple experts' segmentations on the same data set would improve the quality of the data dramatically as this would reduce the impact human error has on the data [23]. Most of the machine learning methods that could use the segmentation data to automate the screening process, however, prefer balanced data sets from which they can learn the characteristics of the data. For this reason, studying the importance of fusing multiple expert segmentations to form a ground truth is important.

1.2 Research objective and scope of the thesis

The objective of this thesis was to get acquainted with and compare the performance of multiple label fusion algorithms. The label fusion algorithms that were used in this thesis were simultaneous truth and performance level estimation (STAPLE) [22], consensus level, labeler accuracy and truth estimation (COLLATE) [1], majority vote (MV), simultaneous truth and performance level estimation with robust extensions (STAPLER) [6] and selective and iterative method for performance level estimation (SIMPLE) [7]. Semi-synthetic segmentations were also generated for this thesis' experiments, to control the characteristics of the virtual experts.

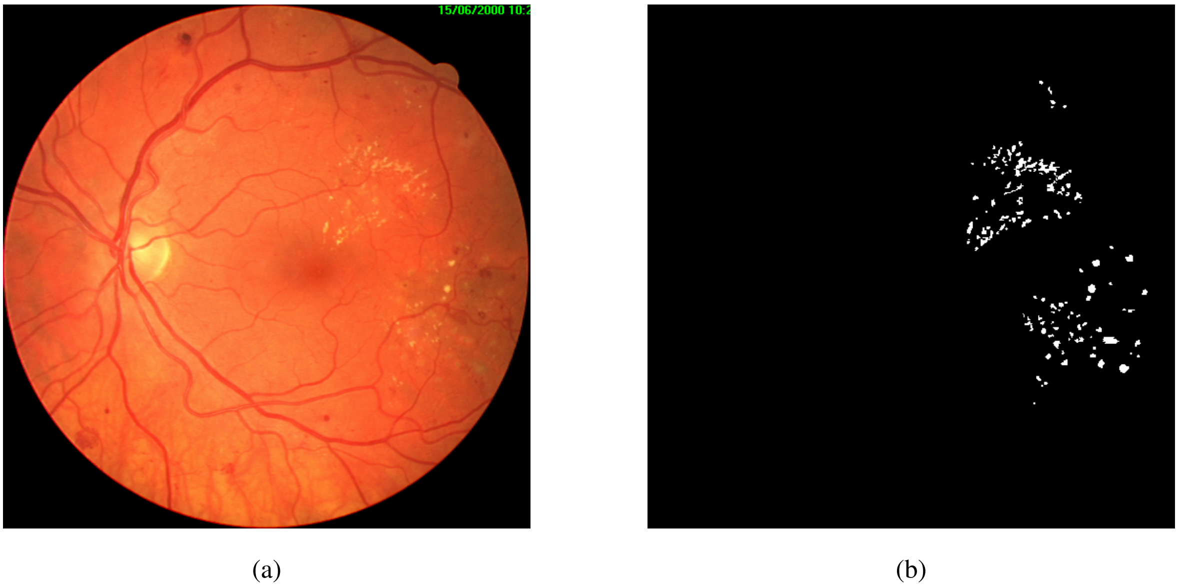
1.3 Structure of the thesis

The next section in this thesis goes more into detail of the material and methods used in this thesis. In section 3 the methodology used in generating the semi-synthetic data is introduced. Section 4 consists of the descriptions of the experiments and their results. Section 5 is reserved for discussion and Chapter 6 concludes the thesis.

2 FUSION OF RETINAL IMAGE SEGMENTATIONS

2.1 Retinal images

The initial data and the ground truth used in this thesis comes from a non-public anonymous Bristol retinal image data set (BristolDB) [17]. The data set contains 107 red-green-blue (RGB) images of size 536x540 and image masks that contain the segments of exudates in the images. Exudates can be a sign of diabetic retinopathy and can cause a treatable loss of vision if present in the macular area of the eye. The image masks found in bristolDB are considered spatially accurate, and the masks have been created manually by a consultant ophthalmologist [17]. An example of a retinal image and an image mask from BristolDB can be seen side by side in Figure 1.



(a)

(b)

Figure 1. (a) Example of a retinal image from BristolDB and (b) example image mask from BristolDB.

2.2 Fusion of manual segmentations

2.2.1 Majority vote

Majority voting is the simplest of the fusion algorithms chosen to be used in this thesis. It is based upon the simple idea of choosing the segment which majority of the observers agree on. The problem with majority voting is that it assumes that all the voters are equally accurate, which is usually not the case when fusing manual segmentations. This can be explained by the voters varying physical capabilities and differences in knowledge [21].

2.2.2 STAPLE

STAPLE is a state-of-the-art label fusion method used nowadays. It is based on the Expectation-Maximization (EM) algorithm [4]. STAPLE algorithm takes a collection of segmentations as its input and produces a probabilistic estimate of the true segmentations and measures the performance level achieved by each observer, which is formed by estimating an optimal combination of segmentations and weighing each segmentation based on the estimated performance level [22].

2.2.3 STAPLER

STAPLER is a very similar algorithm to STAPLE, as it also utilizes the EM-algorithm in its decisions. However, STAPLER, unlike STAPLE, can be run with missing labels, repeated labels, and training trials. Repeated labels in this context means that the observers can generate multiple segmentations on the same image. The missing label feature allows the observers to only partially segment the data to speed up the segmentation process. STAPLERS training trials is a feature, which affects the variability of individual raters by comparing the segmentations with the ground-truth [6].

2.2.4 SIMPLE

SIMPLE also uses an iterative method for determining the resulting segmentations, but unlike STAPLE, it is not based on the EM-algorithm. In SIMPLE, the performance of the input segmentations and the resulting segmentations are estimated in an alternating fashion. SIMPLE, unlike STAPLE, leaves out poorly performing segmentations from the next iterations of the algorithm so that they do not contribute to the results of the algorithm [7].

2.2.5 COLLATE

While the other fusion methods introduced in this thesis use statistical models to estimate raters performance (excluding majority voting), they do not take into account the spatial difference in the data. COLLATE does that, and the reasoning behind it is simple: some regions of the image are harder to segment while others can be obvious. COLLATE estimates the confusion and consensus levels of each segmentation, both of which characterize the likelihood that the observer makes a mistake at a given region. This means that COLLATE is able to make estimates of the observer behavior over different parts of the image [1].

2.2.6 Summary

A number of experiments have already been conducted, whose aim have been to compare different fusion algorithms. One of these is found in a paper by Xu et. al [24]. In this paper, the performance of COLLATE was compared to that of STAPLE. The experiment was done on synthetically generated data, that arguably favored COLLATE as the data was generated in a way that certain regions of the data would be harder to label than others. Never the less COLLATE outperformed STAPLE in said experiments by a clear margin [24]. The data generated for the experiment in the paper is, however, in no way comparable to the retinal image data used in this thesis.

In an article by Commonwick et. al [3], an experiment whose purpose was to validate a label fusion algorithm and compare it to several state-of-the-art label fusion algorithms was conducted. The comparison was done on brain MRI-segmentations, so they differ largely from retinal images. However, the comparison included all the label fusion algorithms used in this thesis [3]. The comparison found that STAPLE, with consensus regions, and COLLATE clearly outperformed all the other label fusion methods used [3]. These results are further supported by another experiment done on brain MRI-segmentations [2], where STAPLE and COLLATE again outperformed the rest of the label fusion methods used in this thesis, while SIMPLE and STAPLER performed even worse than Majority Vote. Using these articles as a reference one could expect COLLATE and STAPLE to outperform the other methods, even on retinal image data, although more recent studies have criticized the usefulness of the results achieved using STAPLE [20].

3 Semi-synthetic generation of retinal image segmentations

Because the data in BirstolDB only includes a single set of segmentations on the images, the need for data to be used as an input for the fusion algorithms arose. For this reason, it was necessary to generate synthetically created data sets using the pixel-wise accurate ground truths and images found in BristolDB. In this section, the steps used in producing the data sets are presented.

The produced data sets were divided into two categories: one that included three synthetic observations and another one that had five observations. The data sets' goal was to simulate the way a real expert would have segmented the images. For the purpose of this thesis, 10 tests were conducted using the semi-synthetically produced data sets with varying input parameters, to experiment with the fusion methods.

The first step in producing the data sets was to convert the original RGB images from BristolDB into CIE 76 L*a*b* colorspace. This was done using the MATLABs rgb2lab function [8]. CIE 76 L*a*b* colorspace was chosen to be used in this thesis as the same amount of numerical difference in CIE L*a*b* colorspace translates to roughly the same visually perceived change detected by humans [19]. After this the images' L*a*b value magnitudes would be determined using the following formula:

$$P_q = \sqrt{L^2 + a^2 + b^2} \tag{1}$$

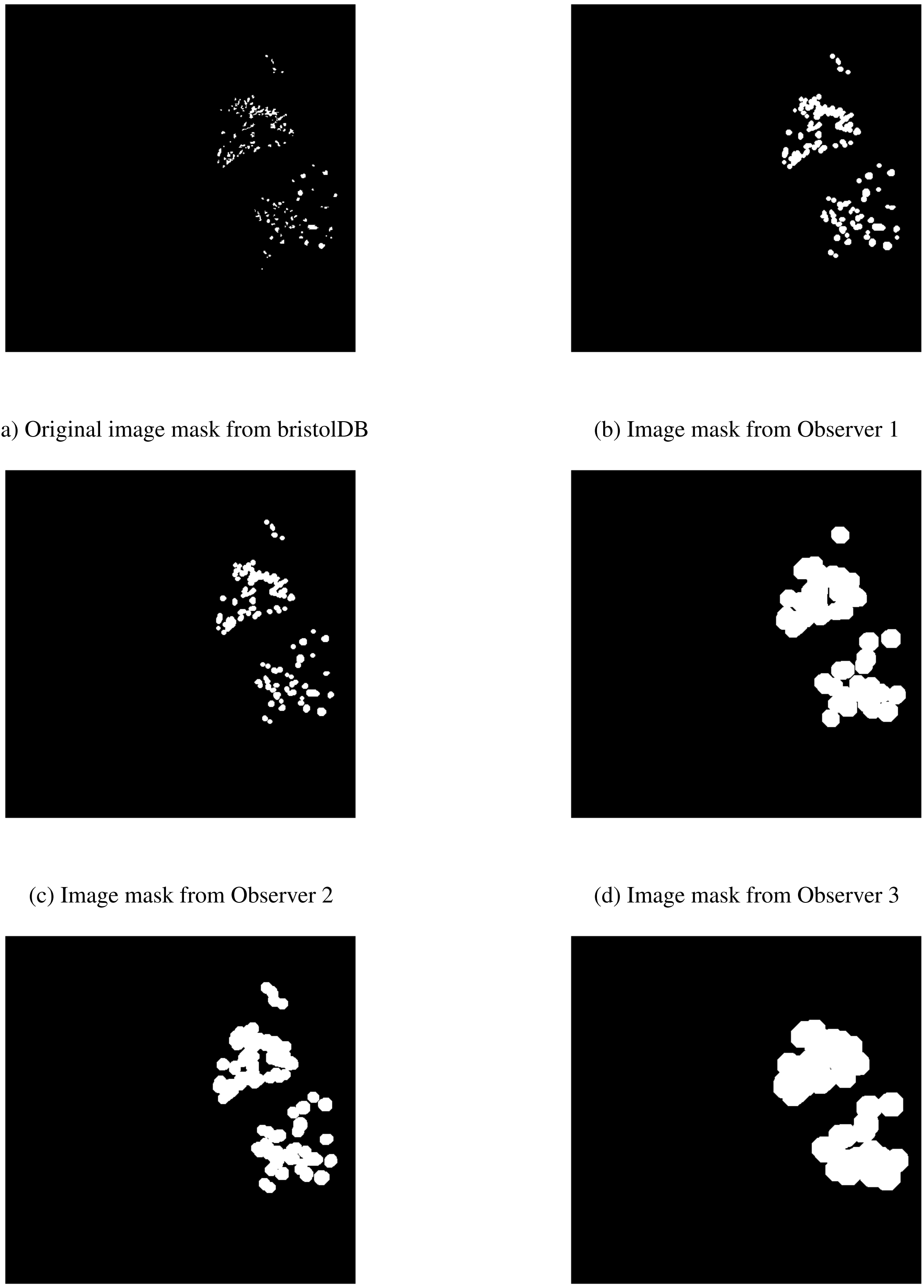
where P_q stands for the magnitude and the L, a and b-values stand for the L*a*b*-values of an image pixel. After this, the P_q-values would be scaled to have values between 0 and 1. The resulting P_q-values are used for modeling the color differences between segmented pixels.

The second step in producing the data sets was to select the characteristics of the so-called virtual observers. These observers would be assigned three different parameters: region-threshold, Mahalanobis distance-threshold and the size of the disk used in dilating segmentations. These parameters would change from one test run to another. The region-threshold was used to remove small segmented regions from BristolDB image masks in an attempt to simulate differences between observers' eyesight, where observers with worse eyesight would fail to segment larger regions than those with, better eyesight. The region-threshold values for each observer are randomly chosen from the closed interval shown in Table 1. Each observer would also be assigned a Mahalanobis distance-threshold value. Mahalanobis distance is a measure that measures the distance between a data point and a distribution [18]. The Mahalanobis distance-threshold value was used to statistically reclassify the segmentations, and each observer would be assigned Mahalanobis distance-threshold value that was generated from a binomial distribution with 20 trials and 0.5 success probability. The Mahalanobis distance-threshold value would then be used to mark segmented pixels as background if their Mahalanobis distance was higher than the threshold and the goal of this was to simulate observers ability to detect the changes in color. The disk size used in dilating would also be selected randomly from an interval shown in Table 1. The goal of this parameter was to simulate the confidence and the precision the observers would have while labeling. More confident and precise observers would have smaller dilating disk sizes, while the less confident and precise observers would have higher dilating disk size, causing their segmentations to be rougher estimates.

The original image masks would be first divided into smaller regions so that connected segmentations would be in the same region. This operation was done using MATLABs bwconncomp function [11]. The virtual observers would fail to label regions that would have a smaller number of pixels than the observers region-threshold, and for each remaining region, a Gaussian mixture model would be determined using MATLABs fitgmdist function [12]. The inputs given to fitgmdist function were the image regions P_q-values and the number of Gaussian mixture components, which was 1 because a more complex mixture was not needed for the distribution. After this, the Mahalanobis distance between the generated Gaussian mixture model and the segmentations would be calculated using MATLABs mahal function [13]. Mahalanobis distances higher than the observers Mahalanobis distance-threshold would be marked as background. After this, the remaining segmentations would be dilated using MATLABs imdilate function [9] with a disk, whose size was determined by the disk size parameter assigned for the observer. After this, the segmentation region would be morphologically opened using MATLABs imopen-function with a disk of size 2 [15]. Finally the borders of the segmentation region would be calculated, and the area filled using MATLABs imfill-function using the ’holes’-parameter [10]. The final step was to combine the newly generated segmented regions into a single image mask. An example of the generated image masks can be seen in Figure 2.

Table 1. The virtual observers used in this thesis. O1-O5 stand for the virtual observers, region-threshold and disk size intervals are the intervals from which the observers are assigned a random value.

	O1	O2	O3	O4	O5
Region-threshold-interval [pixels]	[5 8]	[6 9]	[12 16]	[8 12]	[14 20]
Disk size-interval [pixels]	[2 4]	[3 7]	[8 12]	[6 9]	[10 16]



(a) Original image mask from bristolDB

(b) Image mask from Observer 1

(c) Image mask from Observer 2

(d) Image mask from Observer 3

(e) image mask from Observer 4

(f) image mask from Observer 5

Figure 2. Examples of the synthetically created image masks with the original image mask shown in (a) and (b)-(f) are the image masks produced by the observers in ascending order.

4 EXPERIMENTS AND RESULTS

4.1 Software

The software used throughout this thesis was MATLAB version 2017b [14]. This means that MATLAB was used in creating the synthetic data sets and in running the fusion algorithms on the synthetic data using MASI fusion-library [16], as it also uses MATLAB as its front-end with the back-end being written in Java.

4.2 Hardware

During this thesis, only one running environment was used. The specifications of the running environment can be seen in Table 2.

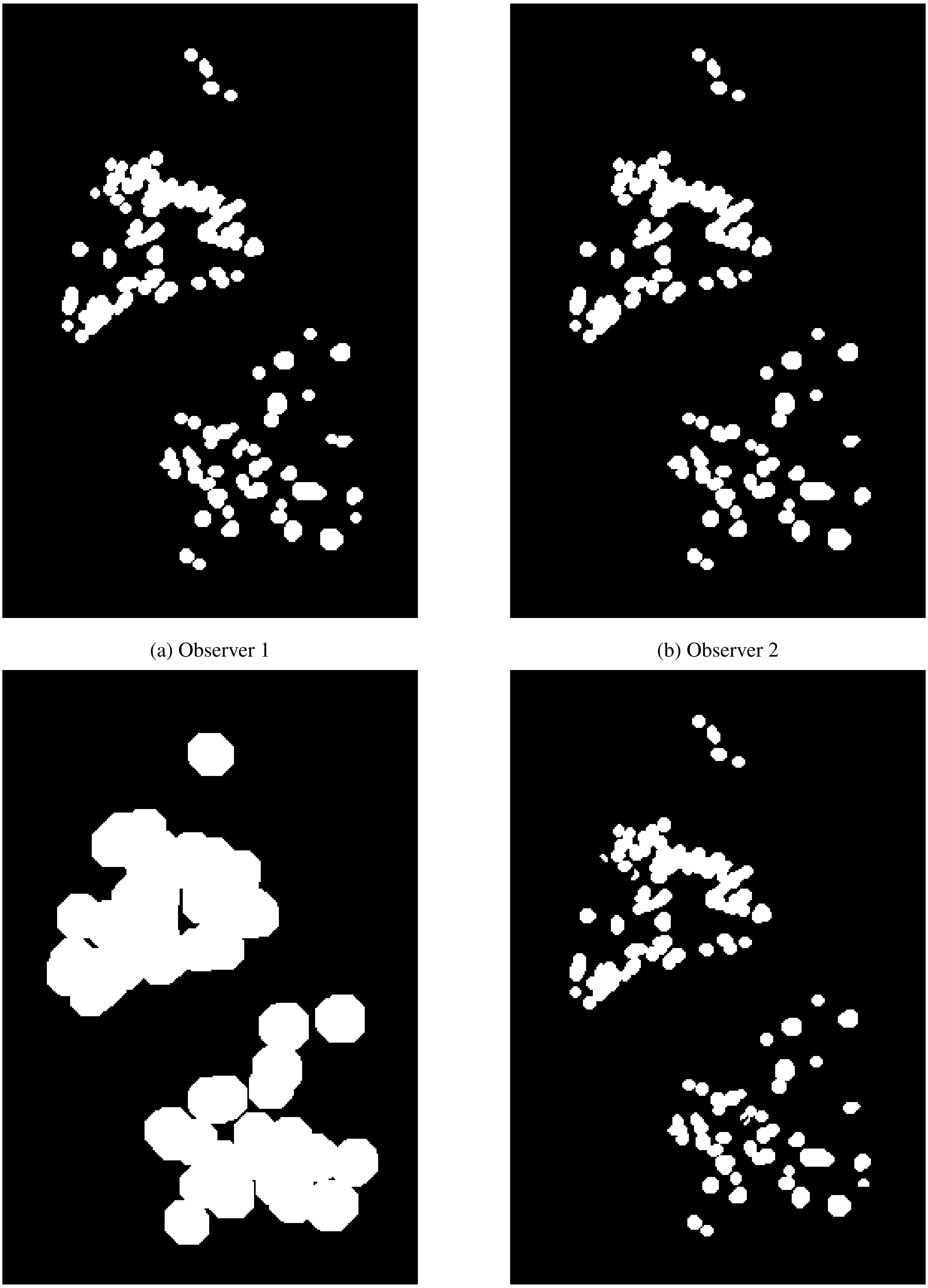
Table 2. Specifications of the running environment

Operating system	Microsoft Windows 10 Pro 64 bit, version 10.0, build 17134
Central processing unit	Intel Core i7-4790K CPU @4.40GHz
Graphical processing unit	NVIDIA GeForce GTX 970 4GB
Memory	16GB DDR3

4.3 Fusion of synthetically created segmentations

4.3.1 Majority vote

Majority vote was implemented in the MASI fusion-library [16] and the implementation was used in this thesis. The example image masks produced by majority voting can be seen in Figure 3 and Figure 4 and they are produced by running the majority voting algorithm with the image masks presented in Figure 2 as inputs.



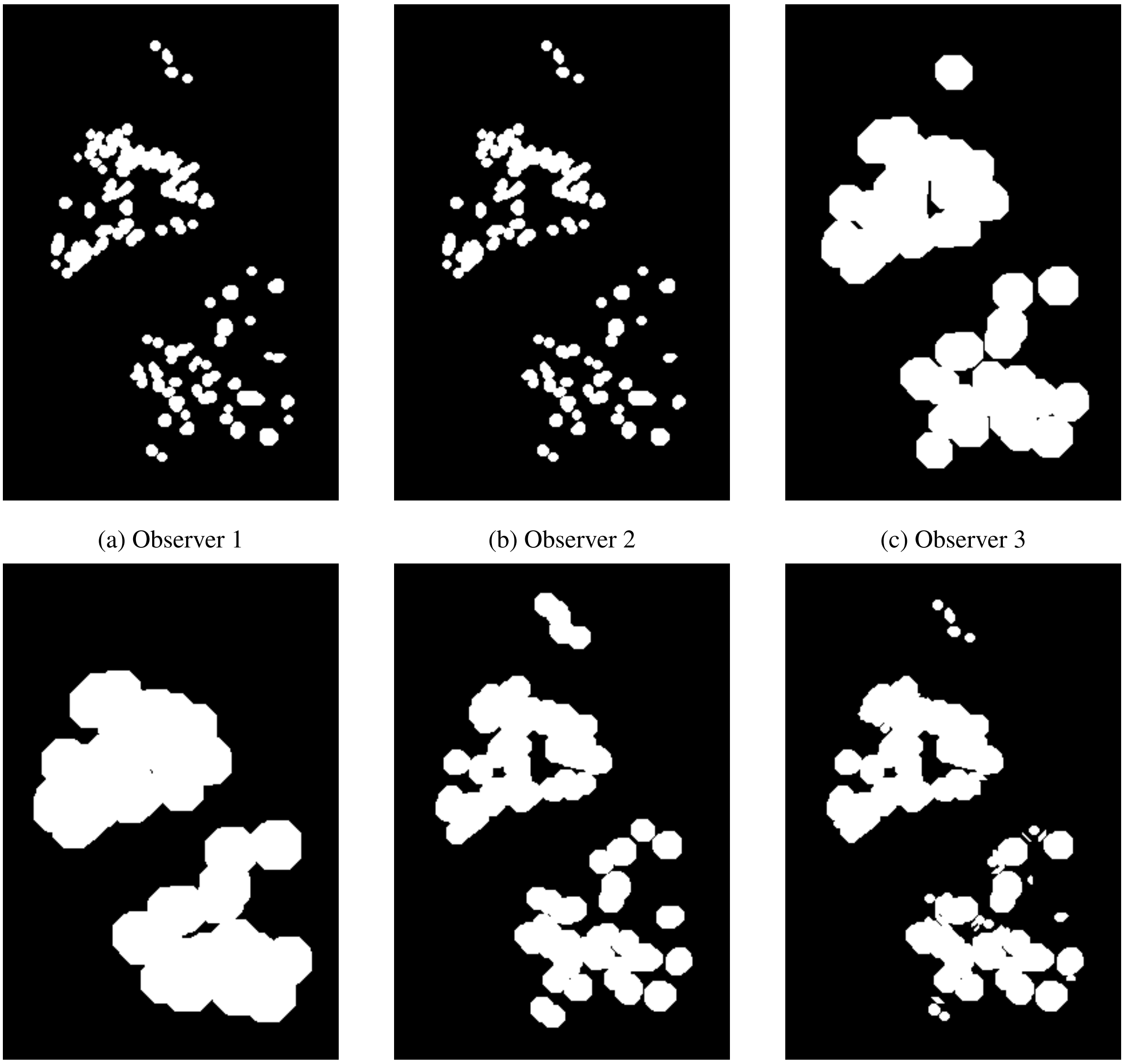
(a) Observer 1

(b) Observer 2

(c) Observer 3

(d) Majority Vote

Figure 3. (a)-(c) image masks from Observers 1 to 3 and (d) example output from Majority Vote on the data set with 3 observers cropped to show the region of interest.



(a) Observer 1

(b) Observer 2

(c) Observer 3

(d) Observer 4

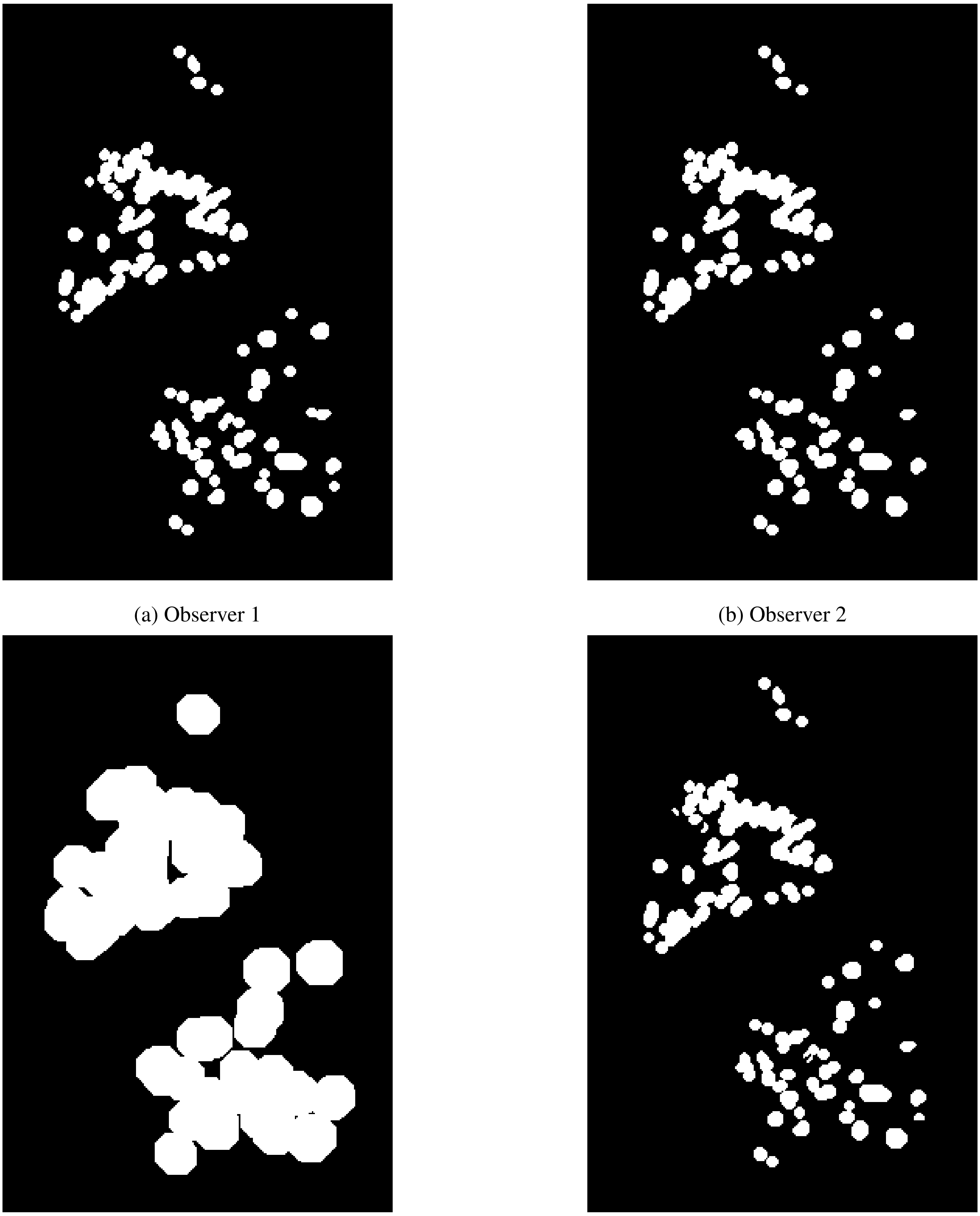
(e) Observer 5

(f) Majority Vote

Figure 4. (a)-(e) image masks from Observers 1 to 5 and (f) example output from Majority Vote on data set with 5 observers cropped to show the region of interest.

4.3.2 STAPLE

In this thesis, the STAPLE algorithm found in the MASI fusion-library was used [16]. The algorithm was used with the ”consensus voxels”-option on, which basically means that the algorithm disregards all the pixels in which all the raters agree on the segmentation and only focuses on pixels that have contradicting segmentations as this is the best performing approach [3]. The other parameters required by MASI-fusion were: epsilon 0.001, init flag 0 and global priors on, as these were the parameters recommended by MASI-fusion in one of their demos [16]. The example image masks produced by STAPLE can be seen in Figure 5 and Figure 6 and they correspond to the image masks presented in Figure 2.



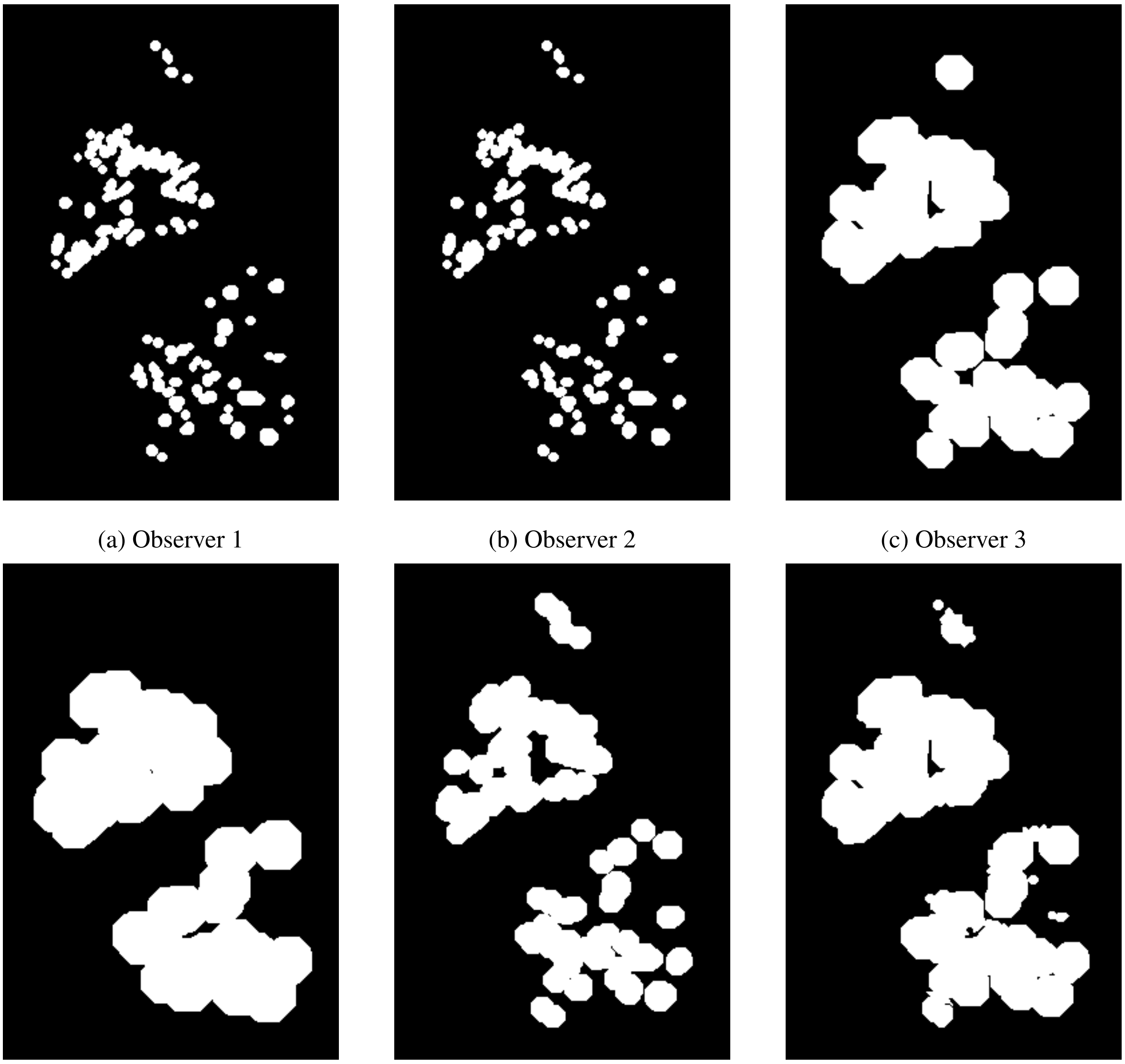
(a) Observer 1

(b) Observer 2

(c) Observer 3

(d) STAPLE

Figure 5. (a)-(c) image masks from Observers 1 to 3 and (d) example output from STAPLE on a data set with 3 observers cropped to show the region of interest.



(a) Observer 1

(b) Observer 2

(c) Observer 3

(d) Observer 4

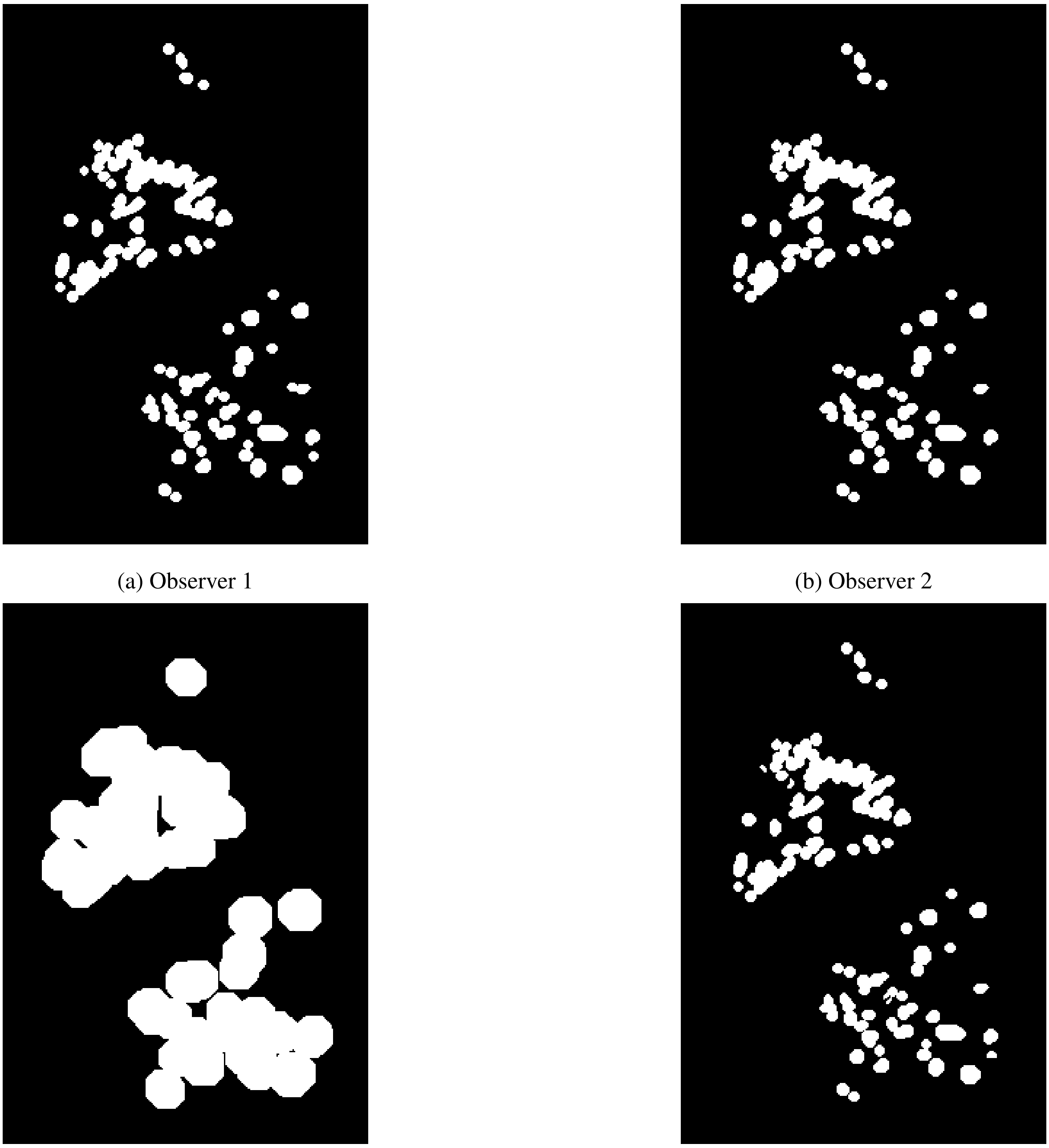
(e) Observer 5

(f) STAPLE

Figure 6. (a)-(e) image masks from Observers 1 to 5 and (f) example output from STAPLE on a data set with 5 observers cropped to show the region of interest.

4.3.3 STAPLER

The STAPLER implementation used in this thesis was also found in the MASI fusion-library [16]. STAPLER was run with the same parameters as STAPLE, and the only exception was that it required the bias-theta parameter as an input that was constructed using MASI-fusions function construct_theta_bias, which took the ground truths found in BristolDB and the synthetically generated data set as its inputs. In this thesis, the _theta_bias was chosen to be constructed with the ground truths found in BristolDB, but in a situation where no ground truths would have been available, they could have been replaced by the ground truths generated by other fusion methods. The example image masks produced by STAPLER can be seen in Figure 7 and Figure 8 and they correspond to the image masks presented in Figure 2.



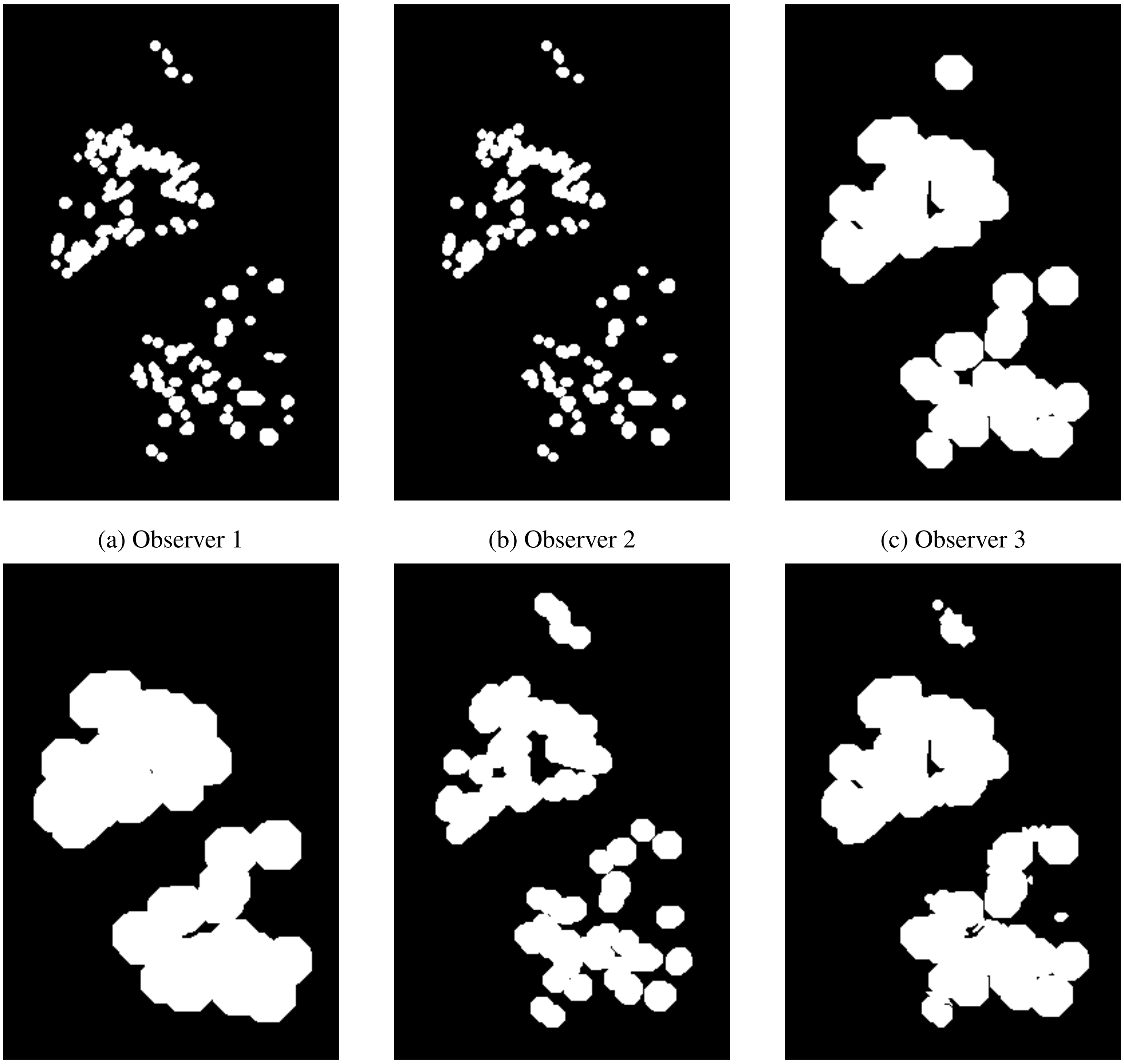
(a) Observer 1

(b) Observer 2

(c) Observer 3

(d) STAPLER

Figure 7. (a)-(c) image masks from Observers 1 to 3 and (d) example output from STAPLER on a data set with 3 observers cropped to show the region of interest.



(a) Observer 1

(b) Observer 2

(c) Observer 3

(d) Observer 4

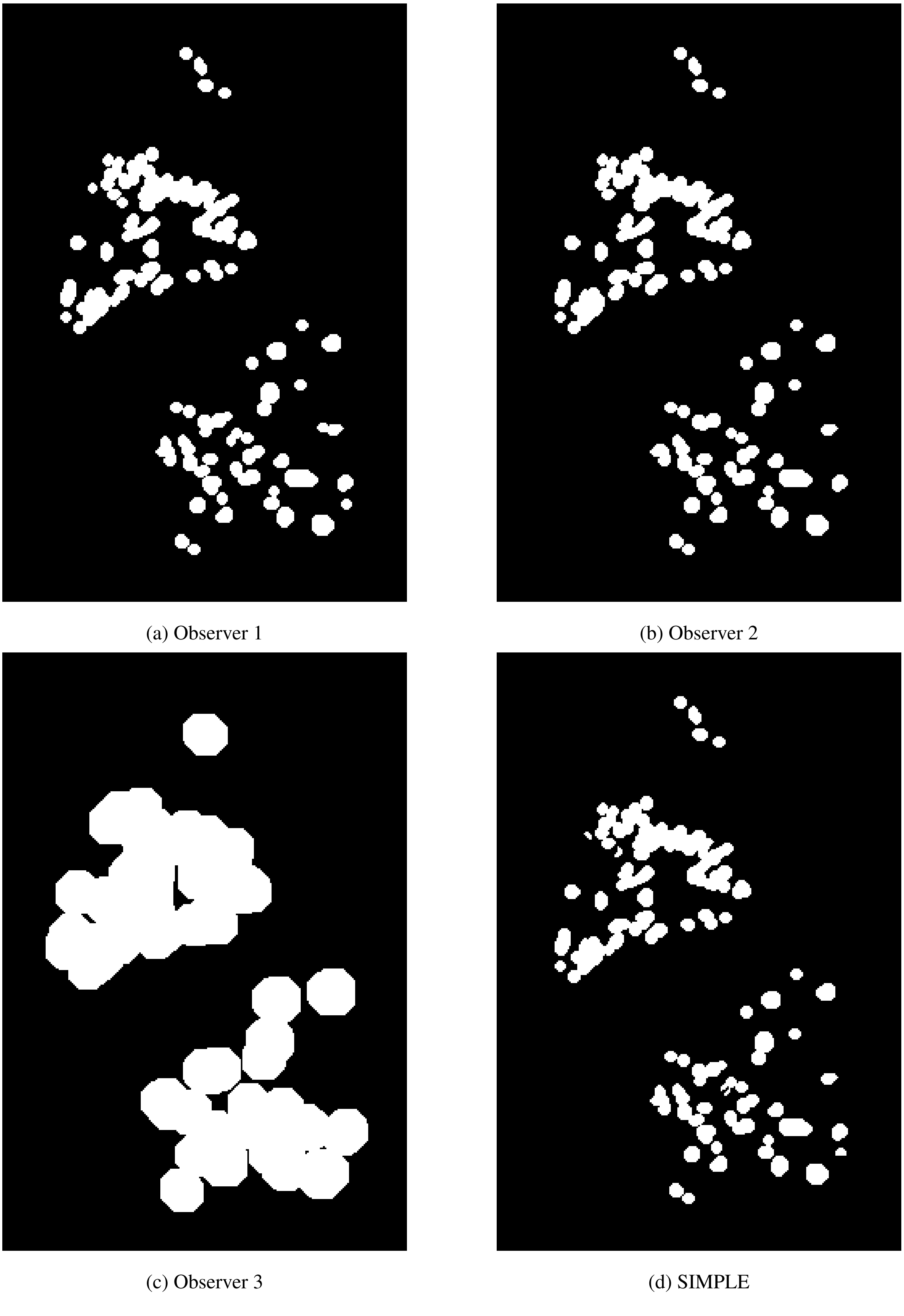
(e) Observer 5

(f) STAPLER

Figure 8. (a)-(e) image masks from Observers 1 to 5 and (f) example output from STAPLER on a data set with 5 observers cropped to show the region of interest.

4.3.4 SIMPLE

In this thesis, the SIMPLE algorithm implemented in MASI fusion-library [16] was used, and it was run with the following input parameters: number of iterations was 3 and performance type was Jaccard index, as this is the performance metric that is used to compare the fusion methods in this thesis. The example image masks produced by SIMPLE can be seen in Figure 9 and Figure 10 and they correspond to the image masks presented in Figure 2.

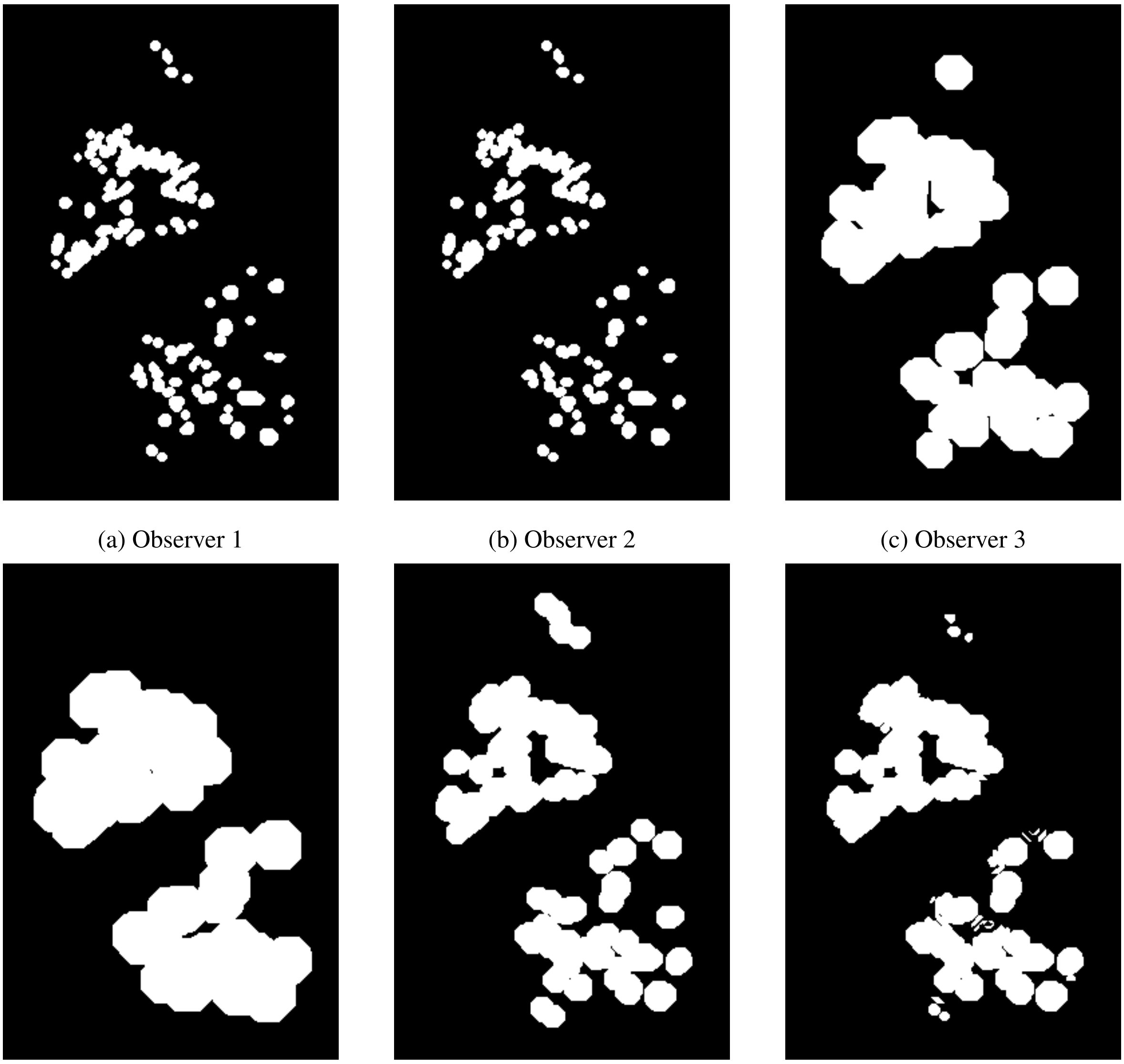

(a) Observer 1

(b) Observer 2

(c) Observer 3

(d) SIMPLE

Figure 9. (a)-(c) image masks from Observers 1 to 3 and (d) example output from SIMPLE on a data set with 3 observers cropped to show the region of interest.



(a) Observer 1

(b) Observer 2

(c) Observer 3

(d) Observer 4

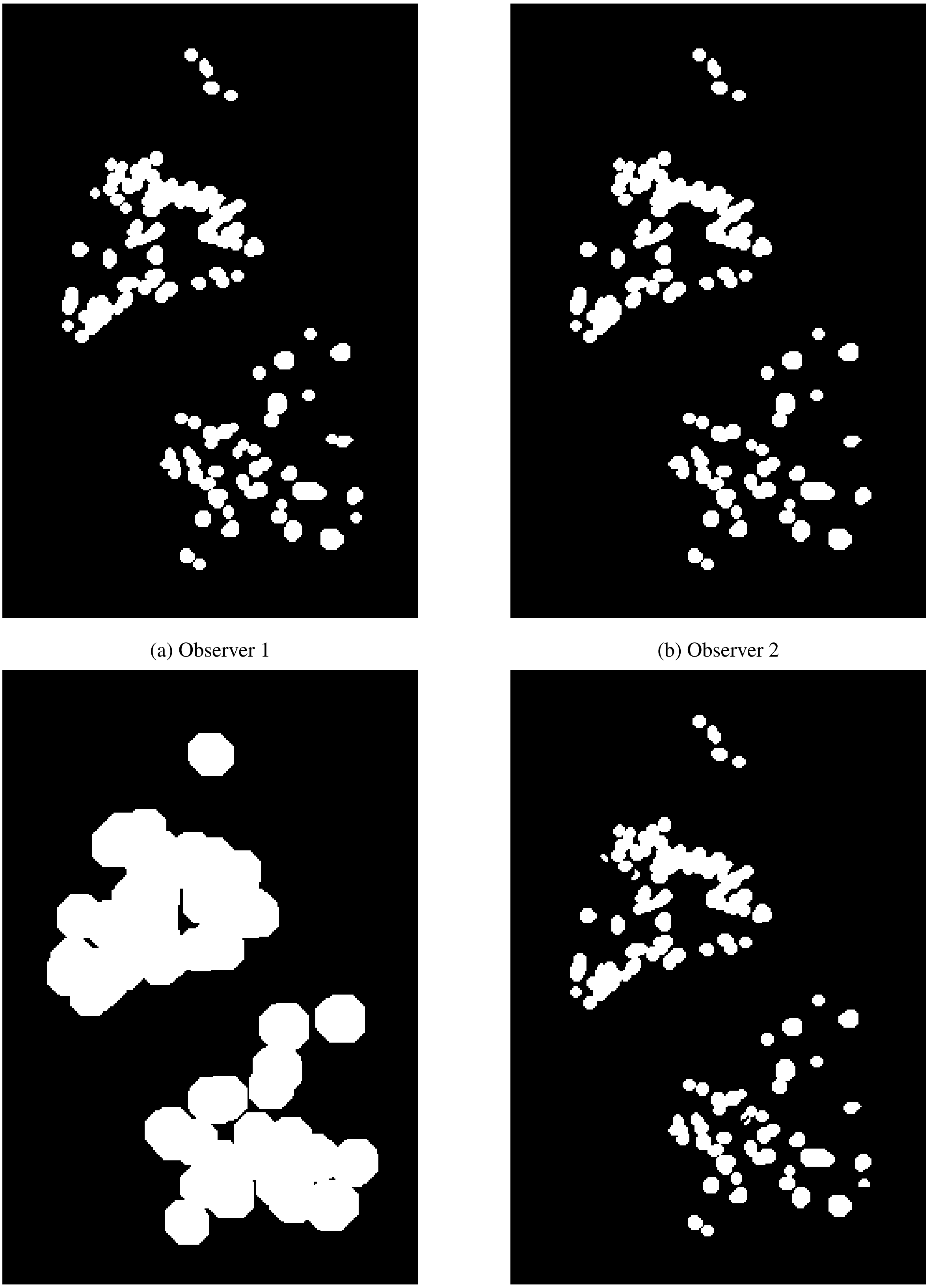
(e) Observer 5

(f) SIMPLE

Figure 10. (a)-(e) image masks from Observers 1 to 5 and (f) example output from SIMPLE on a data set with 5 observers cropped to show the region of interest.

4.3.5 COLLATE

The COLLATE implementation used in this thesis was can also be found in MASI fusion-library [16]. The algorithm was run with the following parameters: epsilon 0.001, init flag 0, prior flag 1, alpha 10^{-3} and cvals 0.95, as they were the parameters suggested by MASI fusion-library [16]. The example image masks produced by COLLATE can be seen in Figure 11 and Figure 12 and they correspond to the image masks presented in Figure 2.



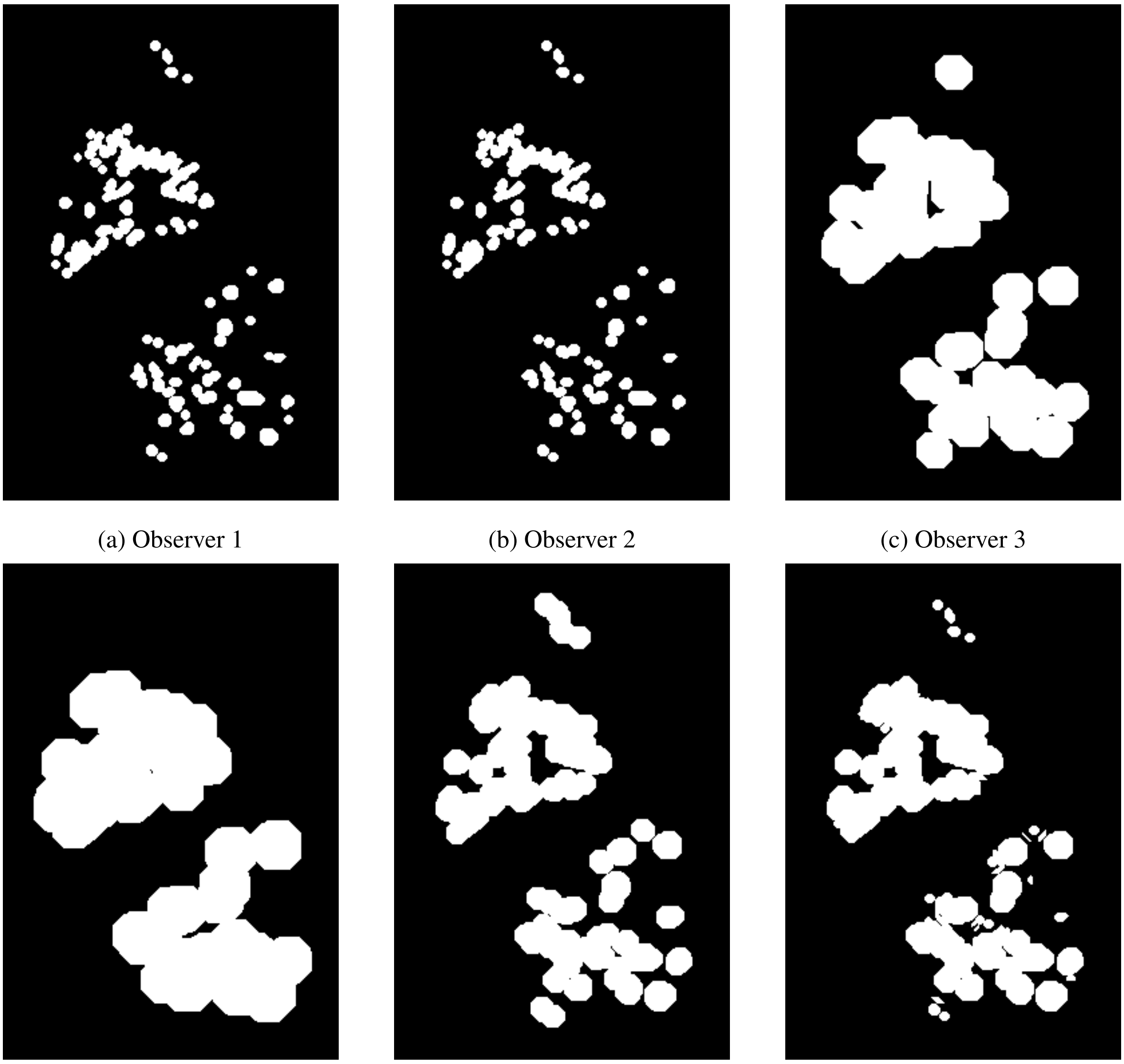
(a) Observer 1

(b) Observer 2

(c) Observer 3

(d) COLLATE

Figure 11. (a)-(c) image masks from Observers 1 to 3 and (d) example output from COLLATE on a data set with 3 observers cropped to show the region of interest.



(a) Observer 1

(b) Observer 2

(c) Observer 3

(d) Observer 4

(e) Observer 5

(f) COLLATE

Figure 12. (a)-(e) image masks from Observers 1 to 5 and (f) example output from COLLATE on a data set with 5 observers cropped to show the region of interest.

4.4 Performance of the label fusion algorithms

The performance metric used in this thesis to compare the fusion algorithms was intersection over union, also known as the Jaccard index. The Jaccard index is a widely used metric in comparing the performance of the segmentations [1]. The values given by the Jaccard index range from 0 to 1, 1 meaning the segmentations being compared are identical. The Jaccard index is calculated as follows:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{2}$$

where A and B are the segmentations being compared and J(A, B) is the Jaccard index between these segmentations. The results of the experiments can be seen in Table 3 for data sets with 3 observers and in Table 4 for data sets with 5 observers as average Jaccard index values from the test cases.

Table 3. Performance comparison of the fusion algorithms over data sets with 3 observers.

	Majority Vote	STAPLE	STAPLER	SIMPLE	COLLATE
Test 1	0.4439	0.4441	0.4439	0.4439	0.4439
Test 2	0.3922	0.3929	0.3922	0.3922	0.3922
Test 3	0.4433	0.4436	0.4433	0.4433	0.4433
Test 4	0.4444	0.4447	0.4444	0.4444	0.4444
Test 5	0.3824	0.3830	0.3824	0.3824	0.3824
Test 6	0.3903	0.3910	0.3903	0.3903	0.3903
Test 7	0.2926	0.2939	0.2926	0.2926	0.2926
Test 8	0.3817	0.3824	0.3817	0.3817	0.3817
Test 9	0.3060	0.3073	0.3060	0.3060	0.3060
Test 10	0.3356	0.3365	0.3356	0.3356	0.3356
Standard deviation	0.05535	0.05497	0.05535	0.05535	0.05535
Average	0.38124	**0.38194**	0.38124	0.38124	0.38124

Table 4. Performance comparison of the fusion algorithms over data sets with 5 observers.

	Majority Vote	STAPLE	STAPLER	SIMPLE	COLLATE
Test 1	0.2783	0.2795	0.2778	0.2783	0.2783
Test 2	0.2783	0.2883	0.2865	0.2783	0.2783
Test 3	0.2654	0.2667	0.2652	0.2654	0.2654
Test 4	0.2813	0.2825	0.2808	0.2813	0.2813
Test 5	0.2663	0.2676	0.2663	0.2663	0.2663
Test 6	0.2808	0.2821	0.2805	0.2809	0.2808
Test 7	0.2874	0.2886	0.2874	0.2874	0.2874
Test 8	0.2729	0.2743	0.2729	0.2729	0.2729
Test 9	0.2874	0.2886	0.2874	0.2874	0.2874
Test 10	0.2877	0.2889	0.2877	0.2877	0.2877
Standard deviation	0.00864	0.00860	0.00862	0.00864	0.00864
Average	0.27945	**0.28071**	0.27924	0.27946	0.27945

5 Discussion

From the Table 3, it can be concluded that the algorithms had very similar performances on data sets with 3 observers, where only STAPLE had a minor edge over the others, and the other algorithms produced the exact same results. However, on data sets with 5 observers, as seen in Table 4, where there was more variation in the data compared to the ground truths, it can be seen that STAPLE was clearly the best performing algorithm and STAPLER performed the worst. It was really surprising how similarly the label fusion algorithms performed, but this could be due to the nature of the synthetically created data.

In future research, it would be beneficial to test the performance of the fusion algorithms on data sets with non-binary data, where other features of the retina would be labeled in addition to the exudates, as well as on larger data sets. Also, the way the synthetic data was produced was not necessarily ideal, and it could be more beneficial to test the algorithms on data that was segmented by actual experts.

6 Conclusion

The aim of this thesis was to compare the performances of different fusion algorithms using retinal image segmentations. Because the initial retinal image data could not be used alone as the input to the label fusion algorithm, two data sets of synthetical segmentations were created. The data sets were created using the BristolDB retinal image database that had segmentations for exudates in retinal images, and the process was documented in detail. The best performing fusion algorithm on this data was STAPLE, although there was no major difference in the performances of the used algorithms.

References

[1] Andrew J. Asman and Bennett A. Landman. “Robust Statistical Label Fusion through Consensus Level, Labeler Accuracy and Truth Estimation (COLLATE).” In: *IEEE Trans Med Imaging* 30.10 (Oct. 2011), pp. 1779–1794. ISSN: 0278-0062. DOI: 10.1109/TMI.2011.2147795. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3150602/ (visited on 03/10/2019).

[2] O. Commowick and S.K. Warfield. “Estimation of Inferential Uncertainty in Assessing Expert Segmentation Performance From STAPLE.” In: *IEEE Transactions on Medical Imaging* 29.3 (Mar. 2010), pp. 771–780. ISSN: 0278-0062, 1558-254X. DOI: 10.1109/TMI.2009.2036011. URL: http://ieeexplore.ieee.org/document/5423294/ (visited on 03/30/2017).

[3] Olivier Commowick, Alireza Akhondi-Asl, and Simon K. Warfield. “Estimating A Reference Standard Segmentation With Spatially Varying Performance Parameters: Local MAP STAPLE.” In: *IEEE Trans Med Imaging* 31.8 (Aug. 2012), pp. 1593–1606. ISSN: 0278-0062. DOI: 10.1109/TMI.2012.2197406. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3496174/ (visited on 04/07/2019).

[4] Joni-Kristian Kamarainen, Lasse Lensu, and Tomi Kauppi. “Combining multiple image segmentations by maximizing expert agreement.” In: *International Workshop on Machine Learning in Medical Imaging*. Springer, 2012, pp. 193–200. URL: http://link.springer.com/chapter/10.1007/978-3-642-35428-1_24 (visited on 03/30/2017).

[5] Joel Kupersmith, Joseph Francis, Eve Kerr, Sarah Krein, Leonard Pogach, Robert M. Kolodner, and Jonathan B. Perlin. “Advancing Evidence-Based Care For Diabetes: Lessons From The Veterans Health Administration.” In: *Health Affairs* 26.2 (Mar. 1, 2007), w156–w168. ISSN: 0278-2715. DOI: 10.1377/hlthaff.26.2.w156. URL: https://www.healthaffairs.org/doi/full/10.1377/hlthaff.26.2.w156 (visited on 04/04/2019).

[6] Bennett A. Landman, John A. Bogovic, and Jerry L. Prince. “Simultaneous Truth and Performance Level Estimation with Incomplete, Over-complete, and Ancillary Data.”

In: *Proc SPIE Int Soc Opt Eng* 7623 (Mar. 12, 2010), 76231N. ISSN: 0277-786X. DOI: 10.1117/12.844182. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2917119/ (visited on 03/10/2019).

[7] T. R. Langerak, U. A. van der Heide, A. N. T. J. Kotte, M. A. Viergever, M. van Vulpen, and J. P. W. Pluim. "Label Fusion in Atlas-Based Segmentation Using a Selective and Iterative Method for Performance Level Estimation (SIMPLE)." In: *IEEE Transactions on Medical Imaging* 29.12 (Dec. 2010), pp. 2000–2008. ISSN: 0278-0062. DOI: 10.1109/TMI.2010.2057442.

[8] MATLAB. *Convert RGB to CIE 1976 L*a*b* - MATLAB rgb2lab - MathWorks Nordic*. URL: https://se.mathworks.com/help/images/ref/rgb2lab.html (visited on 03/11/2019).

[9] MATLAB. *Dilate image - MATLAB imdilate - MathWorks Nordic*. URL: https://se.mathworks.com/help/images/ref/imdilate.html (visited on 06/06/2019).

[10] MATLAB. *Fill image regions and holes - MATLAB imfill - MathWorks Nordic*. URL: https://se.mathworks.com/help/images/ref/imfill.html (visited on 06/06/2019).

[11] MATLAB. *Find connected components in binary image - MATLAB bwconncomp - MathWorks Nordic*. URL: https://se.mathworks.com/help/images/ref/bwconncomp.html (visited on 06/06/2019).

[12] MATLAB. *Fit Gaussian mixture model to data - MATLAB fitgmdist - MathWorks Nordic*. URL: https://se.mathworks.com/help/stats/fitgmdist.html (visited on 06/06/2019).

[13] MATLAB. *Mahalanobis distance - MATLAB mahal - MathWorks Nordic*. URL: https://se.mathworks.com/help/stats/mahal.html (visited on 06/06/2019).

[14] MATLAB. *MATLAB - MathWorks*. URL: https://se.mathworks.com/products/matlab.html (visited on 03/11/2019).

[15] MATLAB. *Morphologically open image - MATLAB imopen - MathWorks Nordic*. URL: https://se.mathworks.com/help/images/ref/imopen.html (visited on 06/06/2019).

[16] NITRC. *NITRC: MASI Label Fusion: Tool/Resource Info*. URL: http://www.nitrc.org/projects/masi-fusion/ (visited on 03/11/2019).

[17] A Osareh, M Mirmehdi, B Thomas, and R Markham. "Automated identification of diabetic retinal exudates in digital colour images." In: *Br J Ophthalmol* 87.10 (Oct. 2003), pp. 1220–1223. ISSN: 0007-1161. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1920779/ (visited on 03/11/2019).

[18] Machine Learning Plus. *Mahalonobis Distance - Understanding the math with examples (python) – Machine Learning Plus*. URL: https://www.machinelearningplus.com/statistics/mahalanobis-distance/ (visited on 06/06/2019).

[19] Schuessler Zachary. *Delta E 101*. URL: http://zschuessler.github.io/DeltaE/learn/ (visited on 03/11/2019).

[20] Koen Van Leemput and Mert R. Sabuncu. "A cautionary analysis of staple using direct inference of segmentation truth." In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2014, pp. 398–406. URL: http://link.springer.com/chapter/10.1007/978-3-319-10404-1_50 (visited on 03/16/2017).

[21] Hongzhi Wang, Jung Wook Suh, John Pluta, Murat Altinay, and Paul Yushkevich. "Optimal Weights for Multi-Atlas Label Fusion." In: *Inf Process Med Imaging* 22 (2011), pp. 73–84. ISSN: 1011-2499. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3226736/ (visited on 04/04/2019).

[22] Simon K. Warfield, Kelly H. Zou, and William M. Wells. "Simultaneous truth and performance level estimation (STAPLE): an algorithm for the validation of image segmentation." In: *Medical Imaging, IEEE Transactions on* 23.7 (2004), pp. 903–921. URL: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1309714 (visited on 05/30/2016).

[23] Gunvor Von Wendt. "Screening for diabetic retinopathy : Aspects of photographic methods." Ph. D. Thesis. Karolinska Institutet, 2005. 62 pp. URL: https://openarchive.ki.se/xmlui/bitstream/handle/10616/39220/thesis.pdf?sequence=1&isAllowed=y (visited on 03/06/2019).

[24] Zhoubing Xu, Andrew J. Asman, and Bennett A. Landman. “Generalized Statistical Label Fusion using Multiple Consensus Levels.” In: *Proc SPIE Int Soc Opt Eng* 8314 (Feb. 23, 2012). ISSN: 0277-786X. DOI: 10.1117/12.910918. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3438516/ (visited on 04/28/2019).

List of Tables

List of Figures