ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 2, Issue 1, January 2013

An Improved Fast Thinning Algorithm for Fingerprint Image

Gayathri S, Dr. V Sridhar
Research Scholar, SJCE Mysore, Karnataka, India, Principal, PESCE, Mandya, Karnataka, India

***Abstract*— A critical step in fingerprint recognition is to skeletonize the fingerprint image for minutiae extraction. This process is referred to as "thinning" in image processing. Thinning is the main preprocessing stage in the fingerprint recognition process. The speed and reliability of the thinning process are important for the whole fingerprint identification system. To accelerate the thinning process, an improved fast thinning algorithm is proposed and implemented in MATLAB and on FPGA.**

***Index Terms*— Fingerprint Recognition, Thinning, Minutiae, FPGA.**

I. INTRODUCTION

Fingerprint image based identity verification is one of the most used biometric systems due to its easy acquisition, high distinctiveness, persistence and acceptance by the public [4]. Usually, the recognition process involves a series of image enhancement and minutiae extraction steps that can be classified as follows:

a. Fingerprint normalization
b. Orientation and frequency image estimation
c. Filtering
d. Image binarization
e. Image thinning
f. Minutiae extraction
g. False minutiae elimination

Among these steps, thinning is one of the most time consuming part. The classical thinning algorithm [3] proposed that the basic operation on each pixel only depends on the value of its 8 neighboring points. As a result, all the pixels in one image could be handled simultaneously. However, due to the natural constraints of the memory size and software coding language, few literatures were reported to fully take advantage of the parallelism of this algorithm. The goal of this paper is to propose a MatLab and FPGA implementation of an improved fast thinning algorithm which exhibits efficient performance.

This paper is decomposed as follows: In section II, we describe the referenced algorithm. In section III we present analysis and Development of proposed algorithm in MatLab and on FPGA environment. Simulation and implementation results are presented in section IV. We present the conclusion in section V. Future scope of the proposed work is enhanced in section VI.

II. REFERED ALGORITHM

The approach described in [1], assumed that a region point in the image has a pixel value '1' and background points have value '0'. The Zhang and Suen' s method consists of successive passes of two basic steps applied for the combination of SI and CGS units, such as current in amperes and magnetic field in oersteds. This often leads to confusion because equations do not match dimensionally. If you must use mixed units, clearly state the units for contour points of the given region, where a contour point is any pixel with value '1' and having at least one 8-neighbor valued '0'. With reference to the 8- neighborhood definition shown in Figure 1, the first step flags a contour point p for deletion if the following conditions are satisfied:

a. $2 \leq N(P1) \leq 6$,
b. $S(P1)= 1$,
c. $P2*P4*P6=0$,
d. $P4*P6*P8=0$,

Where N (Pl) is the number of nonzero neighbors of pi.

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 2, Issue 1, January 2013

That is, N (P1) = P2 + P3 +……..+ P9
And S (P1) is the number of 0-1 transition in the ordered sequence of p2, p3, ..., p8, p9.
For example, N(P1) = 4 and S(P1) = 3 as shown in Figure 1.

P9	P2	P3
P8	P1	P4
P7	P6	P5

0	0	1
1	p_1	0
1	0	1

Fig 1 Neighborhood Arrangement

In the second step, conditions (a) and (b) remain the same, but conditions (c) and (d) are changed to:
c′. P2*P4*p8=0,
d′. P2*P6*P8=0,

Step 1 is applied to every border pixel in the binary region under consideration. If one or more of the conditions (a) through (d) are violated, the value of the point in question is not changed. If all conditions are satisfied the point is flagged for deletion. It is important to be considered, that the point is not deleted until all border points have been processed. This prevents changing the structure of the data during execution of the algorithm. After step 1 has been applied to all border points, those that were flagged are deleted, changed to '0' Then, step 2 is applied to the resulting data in exactly the same manner as step 1.

It is clear that one iteration of the thinning algorithm consists of following steps
(1) Applying step 1 to flag border points for deletion;
(2) Deleting the flagged points;
(3) Applying step 2 to flag the remaining border points for deletion;
(4) Deleting the flagged points.

This basic procedure is applied iteratively until no further points are deleted, at which time the algorithm terminates, yielding the skeleton of the region. Condition (a) is violated when contour point P1 has only one or seven 8-neighbors valued '1'. Having only one such neighbor implies that P1 is the end point of a skeleton stroke and obviously should not be deleted. If P1 had seven such neighbors and it was deleted, this would cause erosion into the region.

Condition (b) is violated when it is applied to points on a stroke one pixel thick. Thus these conditions prevent disconnection of segments of a skeleton during the thinning operation. Conditions (c) and (d) are satisfied simultaneously by the following minimum set of values:
p4 = '0', or p6 = '0', or (p2 = '0' and p8 = '0')
Thus with reference to the neighborhood arrangement in Figure 1, a point that satisfies these conditions as well as conditions (a) and (b), is east or south boundary point or northwest corner point in the boundary. In either case, P1 is not part of the skeleton and should be removed. Similarly, conditions (c') and (d') are satisfied simultaneously by the following minimum set of values:
p2 = '0', or p8 = '0', or (p4 = '0' and p6 = '0')
These correspond to north or west boundary points, or a southeast corner point. Note that northeast corner points have p2 = '0' and p4 = '0' and thus satisfy conditions (c) and (d), as well as (c') and (d'). This is also true for southwest corner points, which have p6 = '0' and p8 = '0'.

III. ANALYSIS AND DEVELOPMENT

Consider an iterative parallel algorithm first presented by Zhang and Suen [1]. Here the algorithm is modified as explained in the previous section and implemented in MATLAB. The pseudo code of the modified fast algorithm uses the following variables. I = the original binary image in which black pixels are “0” and white pixels are “1”. The object in the image is made up of connected white pixels. J and K are temporary images for use in each iteration of the algorithm. J is the $(n-1)^{th}$ iteration output and K is the current, or n^{th} iteration output. P (i) is the current pixel under consideration. The 8-neighborhood around the pixel P1 is shown Figure.2

P9 (i-1,j-1)	P2 (i-1,j)	P3 (i-1,j+1)
P8 (i,j-1)	P1 (i,j)	P4 (i,j+1)
P7 (i+1,j-1)	P6 (i+1,j)	P5 (i+1,j+1)

Fig 2 The 8-Neighborhood of Pixel P1

0	0	1
0	P1 (i,j)	1
1	0	1

Fig 3 Calculation of B

Two other variables used in the algorithm are N and S. N of pixel P1 is defined as: SUM (P2…P9). S of pixel P1 is the count of the number of 0 to 1 transitions in a clockwise circle from P9 back to itself. For example suppose we have the pixels shown in Figure.3. S is equal 2 since there are two 0 to 1 transitions in a clockwise order.The algorithm runs in 2 sub-iterations. During each sub-iteration different rules are applied for deciding whether or not a pixel will be deleted. The pseudo code is as follows:

```
I= original image
J=temporary image
K=temporary image
J=I
K=I
Loop until no pixels are deleted
   //first sub-iteration
For all pixels J (i,j)
Delete K(i,j) if all of the following are true:
a) 2<= N(i,j) <= 6

b) S(i,j)=1
c) P4=1 or P6 = 0 or
      (P2=1 and P8=1)
end
   //second sub-iteration
   for all pixels J(i,j)
Delete K(i,j) if all of the following are true:
a)                    2 <= N(i,j) <= 6
b)                    S(i,j)=1
c)                    P2=1 or P8 =1 or
(P4=1 and P8 = 1)
   end
   J=K
   End
```

IV.RESULTS

Thinning is the main preprocessing stage in the fingerprint recognition process. The algorithm is developed and tested in MATLAB, and then the same algorithm is created in VHDL using Xilinx tools. The result of the thinning stage is very important for the next stage of finger print recognition process that is minutiae extraction. The algorithm is implemented in MatLab and executed. The simulation results of Xilinx ISE and Modelsim is depicted in Figure 4. The original input fingerprint image and the thinned output images are shown in Figure 4 (a) and Figure 4 (b) respectively.

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 2, Issue 1, January 2013

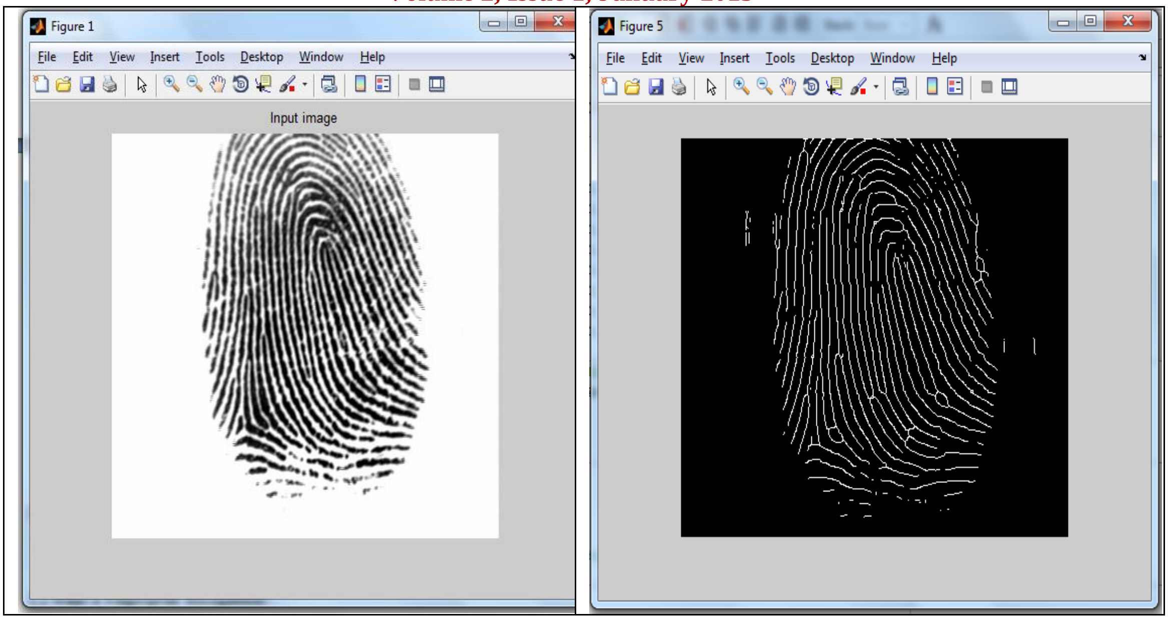


Fig 4 (a) Original image **Fig 4 (b) Thinned image**

Fig 4 Simulation results in Xilinx ISE and ModelSim

VHDL test bench is created in Xilinx ISE to get the simulation results in *MODELSIM*. Here separate test benches are created for binarization and thinning modules. A text file of 1D array of pixel values of the input fingerprint image is taken as input in both the cases. Figure 5 shows the simulation results of MODELSIM output for the binarization module and Figure 6 for the thinning module.

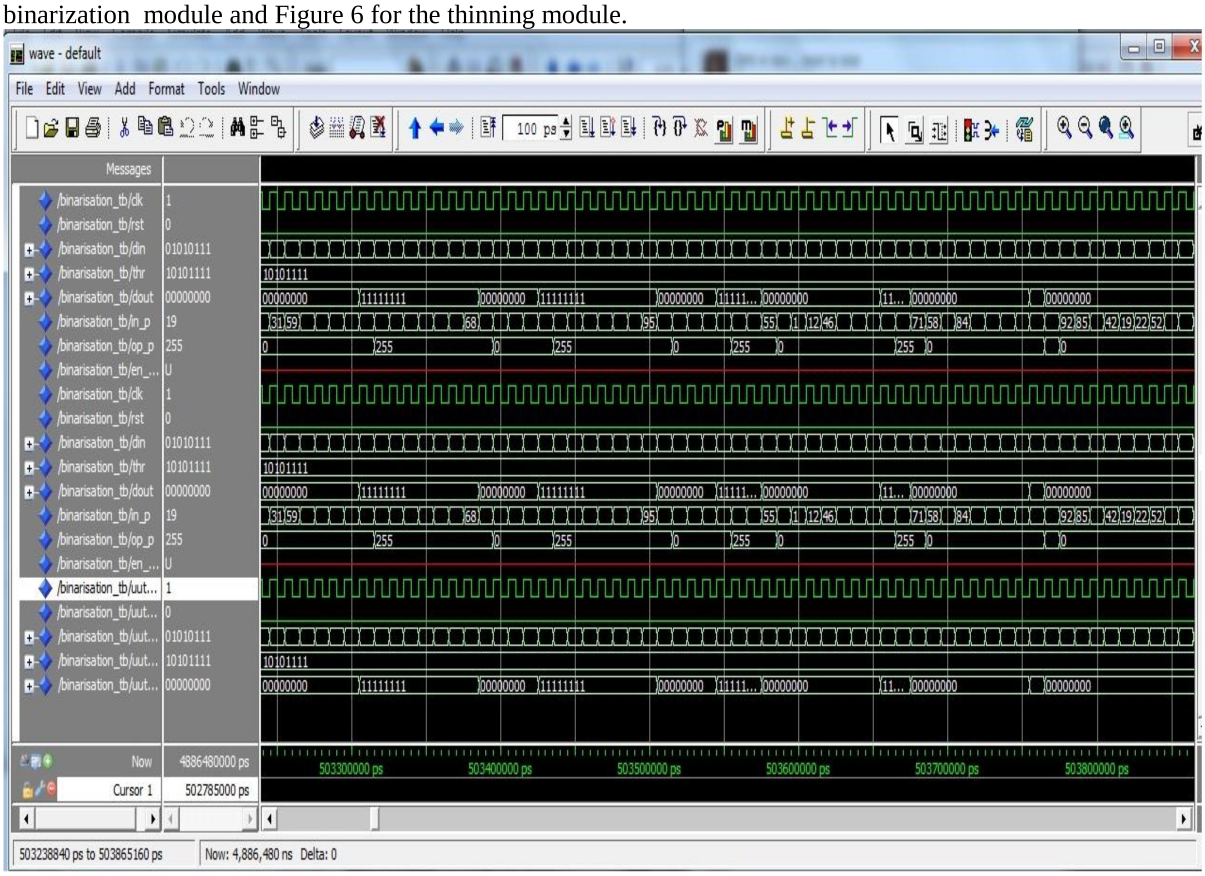


Fig 5 **Simulation Waveform for the Binarization Module**

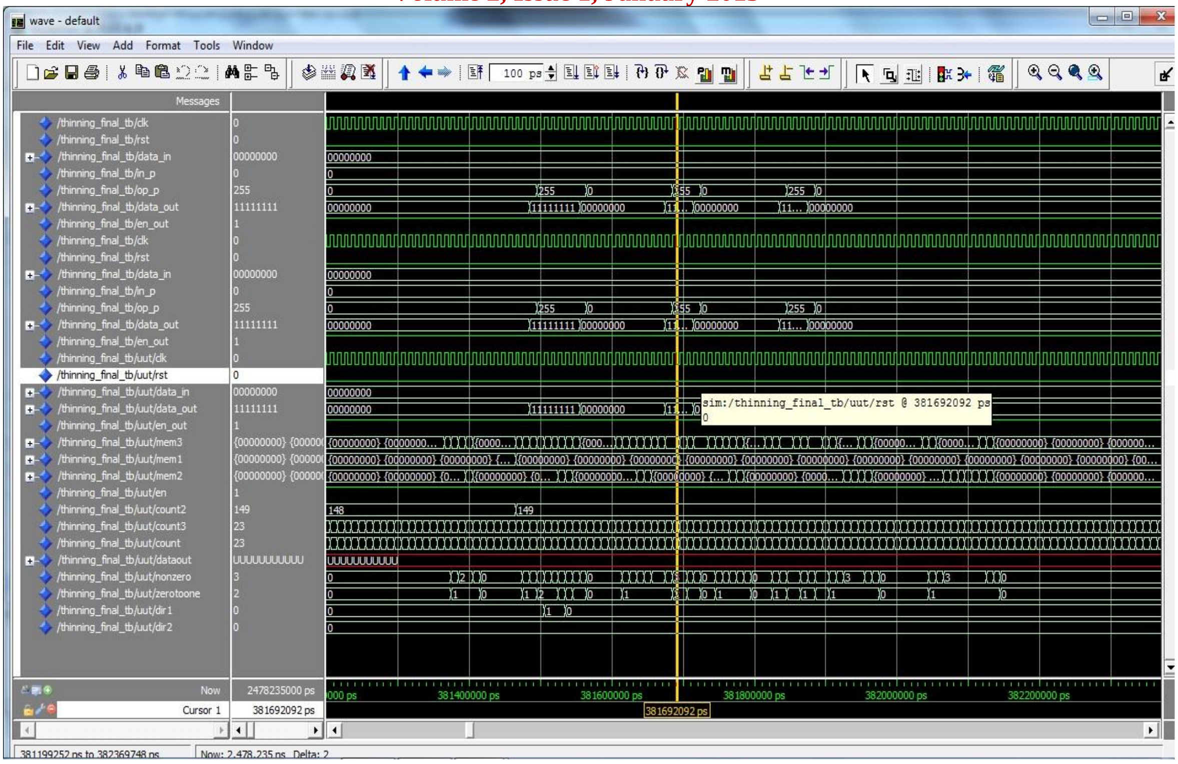


Fig 6 Simulation Waveform for Thinning Module

VHDL outputs

The input 1D text file is processed and the output text file with processed pixel values is obtained by the Xilinx's tool. This text file is converted to image format using MATLAB. Figure 7 shows the output image obtained after the execution of binarization and thinning module.

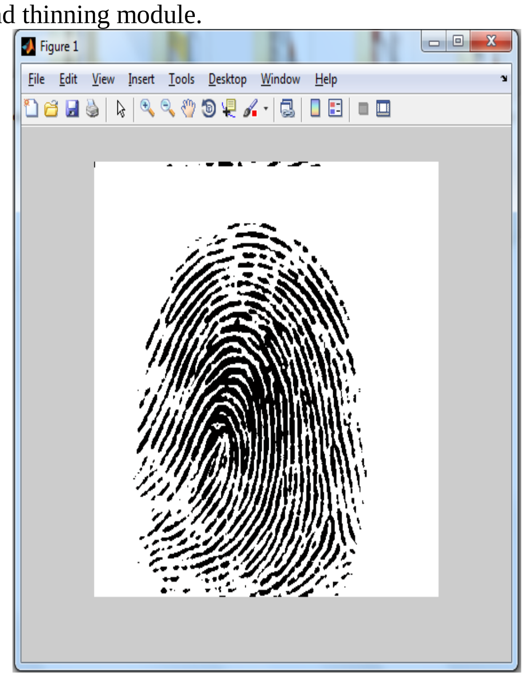
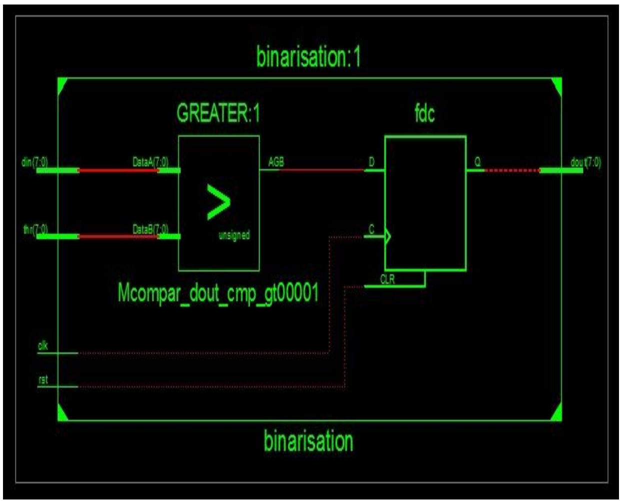


Fig 7 Output Image of the Hardware System

RTL Schematics

The RTL schematics are generated in Xilinx ISE tool as shown in the Figure 8. The RTL schematic for binarization module is as shown in Figure 8 (a) and that of thinning module in Figure 8 (b)

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 2, Issue 1, January 2013



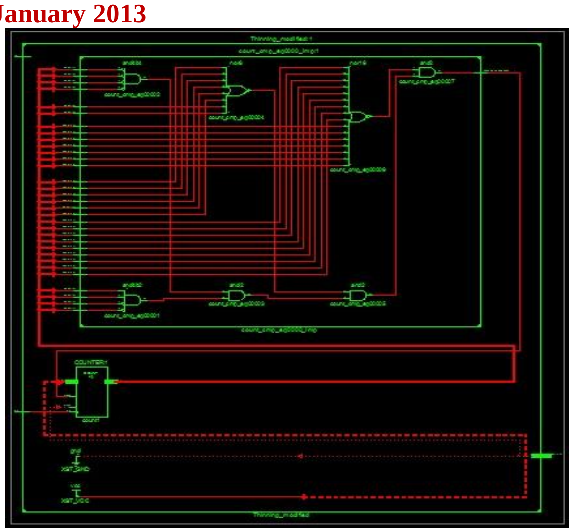


Fig 8 (a) Binarization module **Fig 8 (b) Thinning module**

Figure 8 RTL schematic generated in Xilinx ISE tool

V.CONCLUSION

In this paper, an improved fast thinning algorithm is proposed for thinning fingerprint images. The algorithm is implemented both in Matlab and on FPGA. Experimental results show that the algorithm is more efficient than the referred algorithm.

VI.FUTURE SCOPE

Fingerprint recognition process consists of series of image enhancement and minutiae extraction processes. Out of seven processes only two processes namely image binarization and image thinning are presented. Further the remaining five processes can be implemented and finally an attempt can be made to integrate all seven processes into one which completes the finger print recognition process.

REFERENCES

[1]. Suen, C.Y. and Zhang, T.Y. “A Fast Parallel Algorithm for Thinning Digital Patterns”. Communications of the ACM. Volume 27, Number 3. March 1984.

[2]. Erin Hastings, 2004, A Survey of Thinning Methodologies, College of Engineering and Computer Science, University of Central Florida,

[3]. Sunny Arief Sudiro, 2005, Thinning Algorithm for Image Converted in Fingerprint Recognition System, National Seminar Soft Computing Intelligent Systems and Information Technology 2005, Universities Kristen Petra, Surabaya,

[4]. A. Jain, L. Hong, S. Pankanti. “Biometric identification,” Communication of the ACM, vol. 43, no. 2, pp. 90-98, Feb. 2000.

[5]. H. Xu, Y. Qu, and F. Zhao, “Fpga based parallel thinning for binary fingerprint image,” IEEE Chinese Conference on Pattern Recognition, pp. 1–4, Nov 2009.

[6]. H. Lingga, S. Sudiro, and E. Wibowo, “Hardware implementation of fingerprint image thinning algorithm in fpga device,” IEEE International Conference on Networking and Information Technology, pp. 187–191, 2010.

[7]. M. U. Akram, A. Tariq and S. A. Khan, “Fingerprint image: pre- and post-processing,” Int. J. Biometrics, vol. 1, no. 1, pp. 63-80, 2008.

AUTHOR BIOGRAPHY

Obtained her B.E. in E&C engg from SJCE, Mysore-University of Mysore, Karnataka, M.Tech from Indian Institute of Technology, Madras, pursuing Ph.D at PET Research centre, PESCE, Mandya ,Karnataka . She is serving as Associate professor in the department of E&C, SJCE, Mysore, Karnatakal since 2002 and her field of academic interests are VLSI Design, Digital Image processing, communication system, Testing and verification, HDL, Low

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 2, Issue 1, January 2013

power design, Analog and mixed mode VLSI design. She has published 3 Research Papers,1 International Journals & 1 national journal. Several Projects have been guided for UG & PG Students.

Obtained his B.E. in E&C engg from PESCE Mandya-University of Mysore, M.E. from Calcutta- Jadavpur University, Ph.D from Centre for Biomedical Engineering. Indian Institute of Technology, New Delhi ,awarded during 1995 and POST DOCTORAL RESEARCH from University of Tenaga National (UNITEN) , Kajang, MALAYSIA from 2000 to 2002.He is serving this institution as Principal since 1st May 2008 and his field of academic interests are VLSI Design, Digital signal processing,Bio medical Engineering & instrumentation and cognitive studies, Computer and Mobile communications. He has published 12 Research Papers,6 International Journals & 1 national journal. Several Projects have been guided for UG & PG Students. Has organized several short courses and work shops.