CSI Executive Committee
(2019-20/21)

President
Prof. A K Nayak
(E) aknayak@iibm.in

Vice President cum President Elect
Mr. R K Vyas
(E) rkvyas05@gmail.com

Hon. Secretary
Dr. Santosh Kumar Yadav
(E) secretary@csi-india.org

Hon. Treasurer
Dr. Durgesh Mishra
(E) drdurgeshmishra@gmail.com

Inmd. Past President (2017-18)
Mr. Sanjay Mohapatra
(E) smohapatra70@yahoo.co.in

Regional Vice-Presidents
Region-I (2019-21)
Mr. Arvind Sharma
(E) rvp1@csi-india.org

Region-II (2019-21)
Md. Shams Raza
(E) s_raza2000@yahoo.com

Region-III (2019-21)
Mr. Jayant Bhide
(E) jayantbhid2000@gmail.com

Region-IV (2018-20)
Er. Nachindra K Behera
(E) nachindrabehera@gmail.com

Region-V (2019-21)
Mr. M. S. Prasadbabu
(E) profmsphabu@gmail.com

Region-VI (2018-20)
Mr. Pradeep Rathi
(E) pjrathi61@gmail.com

Region-VII (2019-21)
Prof. Dr. M Sundaresan
(E) bs.sundaresan@gmail.com

Region-VIII (2019-21)
Mr. Konatham Mohan Raidu
(E) kmraidu@informaticsindia.com

Division Chairpersons
Division-I (2019-21)
Mr. Manas Ranjan Pattnaik
(E) manas@anthemgt.com

Division-II (2018-20)
Col. Balraj Anand
(E) b_anand6@rediffmail.com

Division-III (2019-21)
Prof. Suresh Chand Tyagi
(E) sct_35_2000@yahoo.com

dctyagi1963@gmail.com

Division-IV (2018-20)
Prof. Vibhakar Mansotra
(E) vibhakar20@yahoo.co.in

Division-V (2019-21)
Dr. Subhash Chandra Yadav
(E) scy123@gmail.com

Nomination Committee (2019-2021)
Arvind M Nayak
(E) amnayak@hotmail.com

Mr. Deepak Sharma
(E) dsharmaji@gmail.com

Dr. R R Deshmukh
(E) rrdeshmukh.csit@bamu.ac.in
Journal Management Committee
Prof. A. K. Nayak  
President, CSI

Dr. D. D. Sharma  
Chairman, Publication Committee

Prof. (Dr.) Ratnadeep R. Deshmukh  
Editor-in-Chief

Dr. Mayank Agrawal  
Editor

Prof. Brojo Kishore Mishra  
Editor

Editorial Board

Editor-in-Chief
Prof. (Dr.) Ratnadeep R. Deshmukh
Department of Computer Science and IT, Dr. Babasaheb Ambedkar Marathwada University, Aurangabad (MS), India

Editors
Dr. Mayank Agrawal  
Gurukul Kangri University, Haridwar

Prof. Brojo Kishore Mishra  
GIET University, Odisha, India

Editorial Board Members

<table>
<thead>
<tr>
<th>Name</th>
<th>University/Institution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prof. S. K. Gupta</td>
<td>IIT, New Delhi</td>
</tr>
<tr>
<td>Prof. Narendra Choudhry</td>
<td>IIT, Indore</td>
</tr>
<tr>
<td>Prof. Pradip Das</td>
<td>IIT Guwahati, Assam</td>
</tr>
<tr>
<td>Mr. MGPL Narayana</td>
<td>TCS, Hyderabad</td>
</tr>
<tr>
<td>Prof. P. Srinivasa Rao</td>
<td>Andhra University, Visakhapatnam</td>
</tr>
<tr>
<td>Prof. Neeru Adlaka</td>
<td>SVNIT, Surat</td>
</tr>
<tr>
<td>Prof. Anupma Marwaha</td>
<td>Sangrur, Punjab</td>
</tr>
<tr>
<td>Dr. Subhash Chandra Yadav</td>
<td>Central University of Jharkhand, Ranchi, Jharkhand</td>
</tr>
<tr>
<td>Dr. Y.P.S. Berwal</td>
<td>Government Engg. College, Nilokheri, Haryana</td>
</tr>
<tr>
<td>Prof. Andrews Shamraj</td>
<td>Mahendra Engineering College, Tamil Nadu</td>
</tr>
<tr>
<td>Prof. G.V. Chaudhry</td>
<td>SRTMU, Nanded</td>
</tr>
<tr>
<td>Prof. S. N. Kakarwal</td>
<td>Dr. B.A.M. University, Aurangabad(MS)</td>
</tr>
<tr>
<td>Dr. Sukhvinder Singh Deora</td>
<td>Maharshi Dayanand University, Rohtak</td>
</tr>
<tr>
<td>Dr. Shubhangi Sapkal</td>
<td>Government College of Engineering, Aurangabad(MS)</td>
</tr>
<tr>
<td>Dr. Narayan Bhosle</td>
<td>IGNTU, Madhya Pradesh</td>
</tr>
</tbody>
</table>

© Computer Society of India™ Copyright Protected.
# CONTENTS

<table>
<thead>
<tr>
<th>Article</th>
<th>Title</th>
<th>Author</th>
<th>Page No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td>Time Spent on Security Activities and their Impact on Vulnerabilities</td>
<td><em>Dr. Daljit Kaur and Dr. Parminder Kaur</em></td>
<td>4</td>
</tr>
<tr>
<td>R2</td>
<td>Classification of Healthy and Diseased Cotton Plant by using Regression Techniques on Hyperspectral Reflectance Data</td>
<td><em>Priyanka U. Randive, Ratnadeep R. Deshmukh, Pooja V. Janse and Rohit S. Gupta</em></td>
<td>8</td>
</tr>
<tr>
<td>R3</td>
<td>Software optimization and parallelization approaches for solving scientific problems in image &amp; data processing</td>
<td><em>Jai G Singla and Kirti Padia</em></td>
<td>13</td>
</tr>
<tr>
<td>R5</td>
<td>Prediction of Long Duration Complex human activity in video</td>
<td><em>Ashwini S Gavali, S.N. Kakarval and D. V. Kurmude</em></td>
<td>22</td>
</tr>
<tr>
<td>R7</td>
<td>Class Level Code Smells: Chernoff Face Visualization</td>
<td><em>Salman Abdul Moiz</em></td>
<td>36</td>
</tr>
<tr>
<td>R8</td>
<td>Paragraph level semantic plagiarism detection for Marathi language</td>
<td><em>Ramesh Ram Naik, Maheshkumar B. Landge, and C. Namrata Mahender</em></td>
<td>42</td>
</tr>
</tbody>
</table>
Preface

This is the second issue of Volume 3. Earlier first issue of Volume 3 was published in April 2017. Thanks to the former Editor-in-Chief Prof. J.K. Mandal, Kalyani University, India for his sincere effort for this journal. The first paper of this issue entitled Time Spent on Security Activities and their Impact on Vulnerabilities highlights an attempt to know the time required to implement the security activities and its impact on the known vulnerabilities. Second paper entitled Discrimination between Healthy and Diseased Cotton Plant by using Regression Techniques on Hyper Spectral Data highlight the use of Regression technique on Hyperspectral data to discrimination between Healthy and Diseased plants. Third paper entitled Software optimization and parallelization approaches for solving scientific problems in image and data processing highlights use of openMP, MPI and CUDA on advanced hardware architecture like multicore workstations, virtual SMP machine and GP-GPU’s devices for solving real life scientific problems. Forth paper entitled Image Based Rain Removal Process for Vision Enhancement highlights the process of image based rain streak process to obtain the vision enhancement from images. Fifth paper entitled Prediction of Long Duration Complex human activity in video highlights the use of sequential representation of human actions for predicting long duration human activity in video. Sixth paper entitled A Hybrid Approach for Anatomical Retinal Structure Segmentation Using Adaptive Fuzzy Thresholding paper highlights the different types of techniques for segmentation and recognition of retinal daisies. Seventh paper entitled Chernoff Face Visualization highlights the method to visualize the smelly classes in a better and effective way using chernoff faces. The last paper entitled Paragraph level semantic plagiarism detection for Marathi language mainly focused on semantic plagiarism detection for Marathi language at paragraph level.

My sincere gratitude to the members of CSI Exec Com for providing me the opportunity to serve as Editor-in-Chief for the CSI Journals of Computing. I received enormous support from immediate past and present office Bearers of CSI India to process this volume. Thanks to the President Prof. A. K. Nayak, Immediate Past President Mr. Sanjay Mahapatra, Hony. Secretary Prof. S. K. Yadav, Dr. D. D. Sharma, Chairman Publication Committee and others members.

This is the first issue, I am editing this Journal. I hereby place record my sincere gratitude to the members of International Editorial Board and the reviewers who have spared their valuable time for reviewing the papers submitted to the Journal and also provide constructive feedback to the authors.

The success of any Journal depends contribution from authors. I thank the author and wish the Journal to attract more articles and establish its impact in the computing field of research. I am thankful to my scholars of the MSL and HCI research laboratories, Dr. B.A.M.U University, Aurangabad (MS) for their support during entire processing.

I hope this issue will be helpful to the researchers of various emerging areas.

Professor (Dr.) Ratnadeep R. Deshmukh
Dr. Babasaheb Ambedkar Marathwada University, Aurangabad (MS)
Editor-in-Chief, CSI Journal of Computing
January 2020
Security is an imperative attribute of every software system but is not sincerely considered by developers while creating it. Due to tremendous competition of rapid development, developers are spending very less time on the implementation of security features in web applications. Thus, they are inculcated with large number of serious vulnerabilities which may lose the reputation of the organization as well as the sensitive data. Adequate awareness and a little time can save a lot. This paper is an attempt to know the time required to implement the security activities and its impact on the known vulnerabilities.

Keywords: Secure Development, Security, Vulnerabilities and Web applications.

1. Introduction

Security is a major and important attribute of every software system, whether it is web based software, mobile app, standalone computer based desktop application or network based software. But many times, it not considered by the developers of the software as functional requirement of the software. It comes under the category of quality attributes and often considered when the product is ready to use. Software is the collection of lines of code (LOC) written using some programming language. The major cause that software cannot withstand security attacks is the vulnerabilities existing in it. The vulnerabilities are similar to holes which may create gate to enter in software and perform malicious activities with it [1]. This makes the software insecure and such software generates risks to the organization to lose its sensitive data.

In general, there are three main approaches that are considered for developing secure software. Among them, ‘Penetrate and Patch’ is the first approach, which means that enforcing patches to fix vulnerabilities in the software. It is one of the most commonly used technique to secure and save applications, but a study depicts that relative cost of fixing defects during production or after the product is ready for the market or has been released is 30 to 100 times more costlier [2]. The second approach focuses on security implementation through external perimeters by securing operational environment. In this approach, external devices like firewalls, intrusion detection systems are used to make a secure software system. These mechanisms of security can save the environment up to some extent but cannot make application resistible to attacks against software implementation and design. The third and comparatively rigorous approach is secure software engineering. The concept behind this technique is to follow well-structured models of software engineering or processes through the phases of Requirement Analysis to Design and Implementation with security concept from the very beginning. Not a single approach is enough to secure software and thus, a hybrid approach can be followed. Nonetheless, the future probable expenditure can be reduced by addressing security during development [2].

This paper is an attempt to know and visualize the impact of security activities on web applications and time spent in implementing these activities. It is organized in six sections where section I gives the brief introduction and section II briefs out the related work done. Section III outlines the common vulnerabilities in web applications along with their corresponding security activities in software development phases. Section IV represents the methodology used for carrying out the work and result are discussed in section V. Finally, section VI concludes the presented research work.

2. Literature Review

The concern of security in web applications has been realized soon after the popularity of the web applications. Attackers always try to locate vulnerabilities and loopholes in the web applications in order to enter into the system. For developing secure web applications, all the aspects of vulnerability and their impact on the security needs to be carefully examined. Lots of research has been done in this
A report from WhiteHat in [3], has shown the statistics of vulnerability assessment from more than 2000 websites under WhiteHat Sentinel management broken down by industry and size of organization. The average website had nearly 13 serious vulnerabilities. In [4], Cenzic has scanned thousands of applications of its customers and gathered information that 96% of web applications have vulnerabilities with a median of 14 per application. Among these vulnerabilities XSS(Cross Site Scripting), Leakage, Authentication and Session Management are most common. Application layer is the main target of the attackers as developers are not eradicating the vulnerabilities but mainly focusing on the functionality implementation.

The vulnerabilities and attacks have been classified by Chavan, S. B., et al. (2013) in [5]. The classification is done respective to the development phases along with the preventive actions. Symantec report in [6] has revealed that for the year 2016, 76% of the total scanned websites are vulnerable and it also shows that 1.1 billion and 563 million user identities (IDs) were stolen in data breaches in the year 2016 and 2015 respectively. The identities stolen in the later year were almost two times the number in 2015. This is regardless the reality that the number of data breaches actually reduced almost two times the number in 2015. This is regardless the number of data breaches actually reduced between 2015 and 2016 from 1,211 to 1,209.

In [7], the top ten list released by OWASP for the year 2017, it has shown injection flaws are still leading in the web applications as in its previous years reports [8,9].

3. Security activities corresponding to Major vulnerabilities in web applications

Software Vulnerability is a weakness or flaw in the software which may change the software behaviour and malicious users or attackers may exploit it [1,10-11]. Vulnerabilities are inculcated in web applications during development and can be eliminated at the same phase of occurrence. These activities are mapped with respect to the vulnerabilities in [1]. To mitigate these common observed vulnerabilities in the web applications, few general security activities are identified during development phases as:

3.1 Security Activities

1. Sanitize/Validate user input by confirming that data is properly typed and does not contain any escaped code.
2. All the inputs should be validated or checked at both client and server side with respect to the type of the data, its length and format
3. Encrypt string with such a mechanism or technique so that all meta-characters are recognized by the database as normal characters.
4. Use forceful whitelist for naming files that limit the character set to be used. Allow only a single “.” character in the filenames and exclude directory separators like “/”, if possible.
5. Use proper output encoding, escaping, and quoting. For any data that is to be displayed on another page, especially the data received from the external inputs, use the appropriate encoding of the non-alphanumeric characters.
6. Simply, don’t trust on the client side input and always enforce a tight checking of the input entered by user at server side.
7. Handle all the unexpected errors or exceptions. For internal server errors, use generic 500 error page.
8. Load the JavaScript files from the trusted sources only and make sure that end users of the web applications cannot control these sources.
9. Use stored procedures with static SQL wherever possible.
10. Avoid using dynamic queries and prefer parameterized queries.
11. Perform Code reviews.
12. All the servers whether web, application or database should be patched and updated with latest version of the operating systems and other necessary software.
13. All default and not required system stored procedures should be deleted.
14. Unnecessary stored procedures/ prepared statements should be Deleted or Disabled.
15. Directory browsing should be disabled unless it is not required. Even in the case where it is necessary, be sure that the listed files do not induce any kind of risk.

4. Research Methodology

To find the time spent on security activities and their impact on web applications, a set of web applications is collected and hosted on client server based environment. The XAMPP web server is used on the server to host the web applications and on the other hand client is using an open source operating system based on Linux-Kali [12], which provides many open source tools for testing the web applications for security. After that hosted web applications are scanned for the vulnerabilities using OWASP- Zed Application Proxy (ZAP) scanner, which is again an open source tool. The results are recorded and the applications are modified by implementing the security activities in web applications in order to remove the vulnerabilities found in them. To verify the impact of activities, web applications are again tested with the ZAP tool and results found are compared with the previous result. Also the time spent on each web application after implementing the activities is...
noted down to see the impact on vulnerabilities.

5. Result and Discussion

The results are collected for a set of 25 web applications before and after the implementation of the security activities and represented in Table 1. VI_original and VI_later shows the total vulnerable instances found in each original web application (as they are received from the developers) and modified web application (by implementing security activities) respectively, when they are scanned using ZAP. Similarly, Cat_original and Cat_later are the number of vulnerable categories detected in web applications before and after the changes respectively. Also, the table shows the time spent on each web application in implementing the security activities in each web application w.r.t the vulnerabilities found in it.

It has been found that total number of vulnerable instances and the vulnerability categories detected in web applications in earlier developed set are high in comparison to the later developed.

The impact of time spent on each web application in implementing the security activities can be seen in the graph plotted in the figure 1, which clearly depicts the reduction in vulnerable instances.

<table>
<thead>
<tr>
<th>S.No</th>
<th>App-name</th>
<th>VI_original</th>
<th>Cat_original</th>
<th>VI_later</th>
<th>Cat_later</th>
<th>Time spent (in hrs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>FD_AppsWorld</td>
<td>134</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>6.25</td>
</tr>
<tr>
<td>2</td>
<td>FD_Bookshop</td>
<td>146</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>3.75</td>
</tr>
<tr>
<td>3</td>
<td>FD_Career</td>
<td>96</td>
<td>6</td>
<td>1</td>
<td>1</td>
<td>6.75</td>
</tr>
<tr>
<td>4</td>
<td>FD_Caterer</td>
<td>163</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>6.95</td>
</tr>
<tr>
<td>5</td>
<td>FD_CntrlStudio</td>
<td>158</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>3.15</td>
</tr>
<tr>
<td>6</td>
<td>FD_Guidance</td>
<td>132</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>4.25</td>
</tr>
<tr>
<td>7</td>
<td>FD_e-course</td>
<td>53</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>3.05</td>
</tr>
<tr>
<td>8</td>
<td>FD_e-comm</td>
<td>132</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>4.75</td>
</tr>
<tr>
<td>9</td>
<td>FD_enrlSys</td>
<td>75</td>
<td>7</td>
<td>45</td>
<td>3</td>
<td>5.75</td>
</tr>
<tr>
<td>10</td>
<td>FD_Socialnw</td>
<td>1547</td>
<td>7</td>
<td>13</td>
<td>3</td>
<td>0.5</td>
</tr>
<tr>
<td>11</td>
<td>FD_Giftshop</td>
<td>247</td>
<td>9</td>
<td>0</td>
<td>0</td>
<td>12.75</td>
</tr>
<tr>
<td>12</td>
<td>FD_Jewellery</td>
<td>88</td>
<td>7</td>
<td>9</td>
<td>5</td>
<td>6.75</td>
</tr>
<tr>
<td>13</td>
<td>FD_Library</td>
<td>495</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>3.75</td>
</tr>
<tr>
<td>14</td>
<td>FD_Musiclive</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>2.75</td>
</tr>
<tr>
<td>15</td>
<td>FD_Taxiosline</td>
<td>90</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>3.25</td>
</tr>
<tr>
<td>16</td>
<td>FD_Payroll</td>
<td>29</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>4.75</td>
</tr>
<tr>
<td>17</td>
<td>FD_onlineTutorial</td>
<td>114</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>5.75</td>
</tr>
<tr>
<td>18</td>
<td>FD_Bakery</td>
<td>116</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>6.25</td>
</tr>
<tr>
<td>19</td>
<td>FD_Shoestore</td>
<td>170</td>
<td>8</td>
<td>4</td>
<td>4</td>
<td>10.25</td>
</tr>
<tr>
<td>20</td>
<td>FD_Sportskart</td>
<td>557</td>
<td>7</td>
<td>67</td>
<td>1</td>
<td>4.75</td>
</tr>
<tr>
<td>21</td>
<td>ED_webFileMgr</td>
<td>151</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>5.25</td>
</tr>
<tr>
<td>22</td>
<td>FD_Institute</td>
<td>103</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>4.75</td>
</tr>
<tr>
<td>23</td>
<td>FD_Foodsite</td>
<td>2002</td>
<td>8</td>
<td>0</td>
<td>0</td>
<td>9.75</td>
</tr>
<tr>
<td>24</td>
<td>FD_Footwear</td>
<td>226</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>5.75</td>
</tr>
<tr>
<td>25</td>
<td>FD_matrimonial</td>
<td>110</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>4.75</td>
</tr>
</tbody>
</table>

Fig. 1: Vulnerable instances detected in web applications

In average, more than 97 percent of the vulnerable instances are eliminated from the web applications just by spending a little more than five hours on each web application. This time also includes the time taken to locate...
the vulnerability, eliminate it and test it again. If the security activities will be taken care during the development process itself, it may take even less time. The time spent on each web application and the percentage of the vulnerabilities reduced is drawn in Fig. 2.

![Fig. 2](image)

**Fig. 2 : Impact of time spent on implementing security**

**Conclusion**

“Knowledge is power only when it is implemented”, resists in the case of security activities in web applications. In today's distributed environment of the Internet, it is not possible to get the 100 percent security by connecting to this insecure web. But, yes it can be achieved up to large extent by not letting these known vulnerabilities to infiltrate in web applications just by spending few hours in implementing the essential security activities during the development process of the web applications. Spending petty time on refining the web applications can slashed the vulnerabilities and also save the lot of money to be spent on the external security parameters.

**References**


**About the Authors**

**Dr. Daljit Kaur** (Membership No. 2010000269) is currently working as Assistant Professor at Lyallpur Khlasa College, Jalandhar, Punjab, India. She completed her Ph.D in Computer Science from Guru Nanak Dev University, Amritsar in 2018. Her research interest includes networks, web security and security implementation in software development life cycle. She can be reached at jeetudaljit@gmail.com.

**Dr. Parminder Kaur** is working as an Associate Professor in the Department of Computer Science at Guru Nanak Dev University Amritsar, India. She has completed her Ph.D. in the field of Software Engineering from Guru Nanak Dev University Amritsar in 2011. She has around 42 research papers in the International/ National Journals as well as Conferences to her credit. Her research interests include Software Engineering, Component-based Systems, Web Engineering, Open-Source Software and Software Security.
Classification of Healthy and Diseased Cotton Plant by using Regression Techniques on Hyperspectral Reflectance Data

Priyanka U. Randive, Ratnadeep R. Deshmukh, Pooja V. Janse and Rohit S. Gupta

Dept. of Computer Science & IT, Dr. Babasaheb Ambedkar Marathwada University, Aurangabad, India.

The objective of research is to analyze the perspective of the hyperspectral remote sensing to detect the diseases effects on the cotton plant and to compare the reflectance values of healthy and diseased leaf. To compare the spectral reflectance values, leaf samples were collected from the cotton field and then sample measurements of diseased and healthy leaves were collected by using ASD FieldSpec4 spectroradiometer. Vegetation indices (VI) were calculated to identify the effects of disease on the cotton plant. Traditional regression techniques viz. Linear Regression (LR) and Partial Least Squares Regression (PLSR) were used. The cross-validated outcomes demonstrated that the occurrence of disease on the cotton could be predicted with 95% and 80% accuracy respectively.

Keywords: Remote sensing, Linear Regression, Cotton, Vegetation Indices, PLSR.

1. Introduction

As we all know, economy of the India is basically depends on agriculture. Amongst all the major picks that are produced in India, cotton (Gossypium sp) is one of the main crops. Every year almost 70 countries are producing 20 million tons of cotton for its valuable use of fiber. India, United State, Pakistan, China, West Africa and Uzbekistan all these countries are producing 75% of cotton over global production, among these India has the first rank to produce cotton [1]. Because of this, we can say cotton is ‘White Gold’ for Indian farmers. Due to some changes in the environment, the cotton plants may get affected due to the various types of diseases like fungal, bacterial, and viral etc. and plants can also be damaged by insects [2]. Manual examination of diseases on plants is challenging due to the lack of experience and eye vision limitation. The traditional method flops many times to recognize disease correctly [3]. Manual observation of symptoms and laboratory method are time consuming and costly too. Due to all these reasons, the noninvasive spectroscopic methods including regression techniques can be useful for plant disease estimation.

1.1 Plant Disease Assessment by Spectroscopy Method:

Remote Sensing is an extensive area of research which is useful for many applications like Forestry, Agriculture, Weather, Biodiversity etc. In Agricultural field, it can be useful for soil content determination, vegetation analysis etc. through remote sensing. Spectroscopy is a non-destructive technique in hyperspectral remote sensing, that can examine the chemical and physical properties of any material which is also beneficial for plant disease estimation closely that comprises plant level and individual leaf level estimation under the organized condition in laboratory to spectroscopic measurement on the field [4]. Plant leaves contain various types of acids like nucleic acid, protein, amino acid, water, and other materials which are the cause for the photosynthesis capability and the final production [5].

1.2 Spectral Characteristics of a Leaf reflectance:

Spectral signature of a leaf as shown in figure 1. The visible region is affected due to the leaf pigments present inside a leaf. Near-Infrared (NIR) region is affected due to the cell structure of leaf and Short Wave Infrared (SWIR) region is
affected due to the water and the leaf biochemical. Absorption shows in the visible region is due to the leaf pigments like chlorophyll, carotenoid, xanthophyll etc. These pigments have the absorption characteristic. So if there is higher absorption found inside the leaf it means these pigments present in high scale. Reflectance increases or decreases inside the NIR region depends on inter-cell spaces, cell layers and cell size. Reflectance inside the SWIR affected due to the amount of water and foliar constituents [6].

1.3 Vegetation Indices:

Vegetation Indices are small algorithms which are useful to get information from the spectral signature of any type of vegetation. It is also possible to study the vegetation properties by converting the reflectance value into a single value [7].

There are number of vegetation indices available for observing various characteristics of vegetation. From so many years the study of vegetation using vegetation indices centered on by combining visible and near-infrared regions reflectance. NDVI (normalized difference vegetation index) and SR (simple ratio) have been mostly used indices in remote sensing for monitoring the vegetation not only from space but also from ground on regional and global scale. Measurements are carried out using the device Fieldspec. MCARI (Modified Chlorophyll Absorption in Reflectance Index), TCARI (Transformed Chlorophyll Absorption in Reflectance Index), NPCI (Normalized Pigment Chlorophyll Index), OSAWI (Optimized Soil-Adjusted Vegetation Index) was used in the study of various diseases on plants [8].

Materials and Methodology

- Leaf Sample Collection
- Spectral Reflectance Measurement of leaf samples
- Calculate Vegetation Indices (Vi)
- Applied Regression Techniques
- Result Analysis

![Fig. 2: Flowchart of Methodology](Image)

2.1 Leaf Sample Collection:

The region selected for the study was Harsul Sawangi (19.95412, 75.36038), Aurangabad, Maharashtra for cotton leaf samples collection. The annual mean temperature of this area is 17 to 33°C. The average annual rainfall of Aurangabad area is 710 mm.

2.2 Leaf samples Spectral Reflectance Measurement:

Cotton leaf samples from the study area were collected such as healthy leaves and diseased leaves. The diseased leaves were having dark black spots on it. These leaves were placed in airtight bags to maintain freshness of leaves fresh. Spectral reflectance of leaf samples were measured within 2 hours from plucking from the plant. Spectral reflectance of leaves was collected using the device Fieldspec4 Spectroradiometer in controlled condition inside the lab. Fieldspec4 Spectroradiometer provides a wide range of reflectance from 350 to 2500 nm, with a sampling interval of 3 nm (from 350 to 1000 nm) to 10 nm (from 1000 to 2500 nm).

Light source provided with device i.e. 50 W quartz halogen lamp. The Distance of the sample from light source kept 40 cm. Height of light source kept 42 cm. Distance between sample and spectral gun was 11 cm. Before taking a spectral signature of the leaf optimization and calibration of the device was done using white reflectance panel. Ten spectral signatures of each leaf samples were collected. Mean spectra of ten spectral signatures are considered for the further analysis. For data visualization and analysis ViewSpecPro Software was used.

![Fig. 3: Healthy vs. Diseased leaf Spectral Signature](Image)

Fig. 3 shows higher reflectance in Healthy leaf and lower reflectance in diseased leaf. In entire range of electromagnetic spectrum i.e. Blue (400 nm-525 nm), Green (525 nm-605 nm), Yellow (605 nm-655 nm), Red (655 nm-750 nm), and NIR (750 nm-1800 nm) higher reflectance found in healthy group of cotton leaf than the diseased group cotton leaf. This is due to damage cell structure in diseased leaves.

2.3 Calculate Vegetation Indices:

Three were selected MCARI, TCARI and NPCI. These Indices were calculated using the following equations.

- MCARI = [(R_{700} - R_{670}) – 0.2*(R_{700} – R_{550})]*(R_{700}/R_{670}) \quad [9]
- TCARI = 3*(R_{700} - R_{670}) – 0.2*(R_{700} – R_{550})*(R_{700}/R_{670}) \quad [10]
- NPCI= (R_{670} – R_{430})/(R_{670} + R_{430}) \quad [11]
2.4 Applied Regression Techniques:

Two Regression techniques are selected for the result analysis viz. Linear Regression (LR) and Partial Least Square regression (PLSR).

2.4.1 Linear Regression (LR):

The Linear Regression basically having two types, Simple Linear Regression and Multiple Linear Regression. Linear Regression (LR) can be used for finding the relationship between the two continues variable. From a continues variable, the one is a predictor or independent variable and the other one is a response or dependent variable. LR is primarily used in statistical relationships. The relationship between two variables can be deterministic if any one variable is accurately stated by the other. Linear Regression widely used as a statistical technique. It is used to make a model that can show the relationship between variables and used to give a prediction about data.

Equation of LR: \( Y = a + bX \)
- \( Y \) = Dependent Variable,
- \( X \) = Independent Variable
- \( b \) = slope of the line
- \( a \) = \( y \) intercept

2.4.2 Partial Least Square regression (PLSR):

Partial Least Square regression is a data analysis procedure mainly used for statistical analysis purpose and it is proposed by World in 1966 [12]. PLSR has generally used in the study of vegetation since it gives an effective way to use hyperspectral information [13, 14]. In earlier studies, it indicated that PLSR has the ability to use complete spectral information and it provides convenient way for monitoring agricultural crop parameter [15, 16].

By using PLS regression, where VIS/NIR spectra represented X-data matrix and the dependent variable (Y) is categorical and represented samples class membership (i.e. \( Y \) with values of 0 and 1, where 0 represents healthy class and 1 represent diseased class).

\[ Y = b_0 + b_1X_1 + b_2X_2 + ... + b_pX_p \]
- \( Y \) - Dependent Variable
- \( X \) - Predictor variable
- \( b_0 \) - regression coefficient for the intercept
- \( b_p \) - regression coefficients for variables 1 through \( p \).

2.5 Result Analysis:

In this study, two regression techniques were used linear regression (LR) and Partial Least Square Regression (PLSR). Three vegetation indices were considered MCARI, TCARI and NPCI. Two classes of leaves were considered Healthy and Diseased represented by numbers 0 and 1 respectively. Using LR and PLSR we have plotted cross-validated predictions in terms of Actual vs. Predicted. Figure 4 shows a Linear Regression plot and Figure 5 shows the Partial Least Square Regression plot.

![Figure 4: Linear Regression Plot](image)
![Figure 5: PLSR Regression plot](image)

Table 1: Regression Results of Healthy and Diseased Cotton Plant leaves on Reflectance Values

<table>
<thead>
<tr>
<th>Regression Techniques</th>
<th>Accuracy(R^2)</th>
<th>RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>LR</td>
<td>0.95</td>
<td>0.01</td>
</tr>
<tr>
<td>PLSR</td>
<td>0.80</td>
<td>0.04</td>
</tr>
</tbody>
</table>

LR shows higher accuracy i.e. R^2 is 0.95 whereas in PLSR shows 0.80 Root Mean Squared Error (RMSE) found higher in PLSR is 0.04 and in LR it is 0.01.

Teena et al. worked on bacterial contamination of spinach leaves used equipment a line-scan pushbroom Hyperspectral imaging system in the range of VIS/NIR (456–950). They applied PLS-DA get 84% accuracy [17].

Xu H. R. et al. observed minor damage on tomato leaf. Five levels of damage to tomato plant have been considered. For database collection, they used Nexus FT-NIR spectrometer. They applied regression techniques and get an overall accuracy of 78.33% [18].

Purcell et al. worked on rice brown planthopper and leaffolder infestation. They used VIS/NIR spectroscopy (350–2400) and device fiber-optic probe attached to Fourier transform infrared spectrometer. They considered two classes healthy and infected rice plants. Using linear regression model they get 92% result (R^2=0.922) [19].
III. Conclusion:

This study demonstrates that it is possible to detect a disease effect on the cotton plants. Regression techniques equally significant for the prediction of disease effect along with the vegetation indices. With hyperspectral techniques, it is possible to achieve fast detection of disease of crops in a non destructive way.

Acknowledgment:

This work has been supported by DST-FIST with sanction number- SR/FST/ETI340/2013. The Authors are thankful to DST-FIST and Department of Computer Science and Information Technology of Dr. Babasaheb Ambedkar Marathwada University, Aurangabad, Maharashtra, India for providing necessary support and backup.

References:


Classification of Healthy and Diseased Cotton Plant by using Regression Techniques on Hyperspectral Reflectance Data

About the Authors

Ms. Priyanka U. Randive received the MCA degree from Govt. College of Engineering Aurangabad, Maharashtra, India, in 2013. She is currently pursuing the M.Phil degree in Computer Science from Department of Computer Science and IT, Dr. Babasaheb Ambedkar Marathwada University, Aurangabad, Maharashtra, India. She is presently working as Research Scholar in the Department of Computer Science and Information Technology, Dr. Babasaheb Ambedkar Marathwada University, Aurangabad. Her research interest includes Remote Sensing and Geographical Information System (GIS) Technology.

Dr. Ratnadeep R. Deshmukh has completed Ph.D. from Dr. B. A. M. University in 2002. He is working as a Professor in Computer Science and Information Technology (CSIT) Department, at Dr. Babasaheb Ambedkar Marathwada University, Aurangabad (MS) INDIA. He is a Sectional President of Information and Communication Science & Technology (including Computer Sciences) section, Indian Science Congress 2019. He is a fellow and Chairman of IETE, Aurangabad Chapter and life member of various professional societies as ISCA, CSI, ISTE, IEEE, IAEng, CSTA,IDES, Etc. He has published more than 165 research papers in various National and International Journals and Conferences.

Ms. Pooja Vinod Janse is currently pursuing the Ph.D. degree in Computer Science and Engineering from Department of Computer Science and IT, Dr. Babasaheb Ambedkar Marathwada University, Aurangabad, and Maharashtra, India. She is currently working as BSR Senior Research Fellow under the project entitled “UGC SAP-II DRS Phase II Biometric: Multimodal System Development” sanctioned by UGC to the Department of Computer Science and Information Technology, Dr. Babasaheb Ambedkar Marathwada University, Aurangabad. Her research interest includes the digital speech signal processing, Remote Sensing and Geographical Information System (GIS) Technology.

Rohit S. Gupta has completed MSC (Computer Science) degree from Dr. B. A. M. University in 2016. He is currently pursuing the Ph.D. degree in Computer Science from Department of Computer Science and IT, Dr. Babasaheb Ambedkar Marathwada University, Aurangabad, Maharashtra, India. He is presently working as DST Inspire Research Fellow in the Department of Computer Science and Information Technology, Dr. Babasaheb Ambedkar Marathwada University, Aurangabad. His research interest includes Remote Sensing, Geographical Information System (GIS) Technology and Satellite Image Processing.
Software optimization and parallelization approaches for solving scientific problems in image & data processing

Jai G Singla and Kirti Padia

Signal & Image Processing Group, Space Applications Centre, Indian Space Research Organization, Ahmedabad
E-mail: {jaisingla, kirtipadia}@sac.isro.gov.in

In recent times, there is high requirement & availability of high resolution data with higher frequency. This leads to growth of data at tremendous rate and there is challenge to process this data in real time scenario to disseminate timely and useful information for benefit of the society. Also many times data are complex in nature which results in increased data volume. In scientific domain, we have encountered many data crunching problems like near real time image processing, solving computation fluid dynamics (CFD) problem at faster speed, computation of value of PI with higher digits’ accuracy level and solving weather forecasting related problems in given stipulated time. In this paper, we are describing real problems faced in scientific domain and approaches, methodologies and tools used to solve these problems effectively and efficiently in a timely manner. We have mainly used concepts of SW parallelization (using OpenMP, MPI and CUDA programming) and optimization on different HW architectures such as multicore machines, virtual SMP and GP-GPU devices to solve problems in optimum time. Technical paper contains three different problem definitions, hardware and software used for implementation, approaches and techniques used for optimization and efficiency / speed up results achieved.

Keywords: CFD, PI, image processing, range compressing, CUDA, vSMP, MPI, openMP

1. Introduction

There are plenty of programming paradigms available for computer programmers to solve real life scientific problems in near real time scenarios. We have used those programming paradigms such as openMP, MPI and CUDA on advanced hardware architecture like multicore workstations, virtual SMP machine and GP-GPU’s devices. Scientific problems have mandatory requirement to maintain accuracy and precision of the results. There are other requirements of simultaneous processing of large amount of data. We have kept these important aspects in designing optimized solutions for all the problems. In subsequent sections, three different types of problems are mentioned along with their implementation details.

2. Problem 1: Processing of Synthetic Aperture Radar (SAR) data in real time

Each target in echo data is spread out across many range samples (equivalent to chirp length) and in the along-track direction (by the real beam moving through the point target for the duration of the dwell time). Thus for a proper SAR Image formation, each target is compressed in range direction followed by a compression in azimuth direction. The target is compressed in the range direction in order to sum up the energy which is spread across many range samples and then it is placed at a given range position in the signal space. The compression of the target in the range direction is termed as the range compression. As depicted in Fig1, Range compression consists of modules of reading the data from a file; convert the data into input form by subtraction, FFT of the input data that is of 8k x 18 K, FFT of replica data, multiplication of FFT of input data and FFT of replica data. Further, IFFT of resultant of multiplication then further writing the scaled output to a file on disk. In the figure H stand for Host and D stands for Device. In this exercise, we have developed range compression module in CUDA for efficient timings. Hardware architecture, optimization techniques used, kernel launch configuration and results and result validation & speed up is listed in subsequent sections.

Fig 1: Range compression flow chart
Software optimization and parallelization approaches for solving scientific problems in image & data processing

2.1 Hardware architecture & tools used:

We have used following GPU architecture and tools for development and execution.

1. “Tesla M2070” GPU device with total amount of global memory: 5375 Mbytes
2. (14) Multiprocessors, (32) CUDA Cores/MP: 448 CUDA Cores, GPU Clock rate: (1.15 GHz)

2.2 Optimization techniques

To implement the code on GPU, first of all major operations were understood. A thorough analysis of major code segments using profiler tools was done. Compute portions and memory transfers operations in the module were segregated. Further, while porting the code on GPU, device architecture, optimized rules were taken into considerations. Following strategies were used for optimizations:

i. Maximized independent parallelism. [5]
iii. GPU’s shared memory is used for parallel processing.
iv. Problem is computationally equally partitioned to keep the GPU multiprocessors equally busy.
v. Used Pinned memory concepts.
vi. Used pipelining concept for data level parallelism.
vii. Used CUFFT library calls for computation of FFT.

2.3 Optimized kernel launch configurations for Instruction Level Parallelism (ILP)

Launch configuration is the number of blocks and number of threads per block, expressed in CUDA with the "<<<>>>" notation:

mykernel<<<num_blocks,threads_per_block>>>(…);

We had the question of What values should we pick for these?
• Need enough total threads to process entire input
• Need enough threads to keep the GPU busy
• Selection of block size is an optimization step involving warp occupancy

To understand this, we used CUDA occupancy calculator tool [7]. We improved upon the initial timing of our module using the basic concepts of occupancy Vs latency trade off. We could improve our results here by launching lesser threads and increased more number of blocks as shown in table 1. As our data is doing a lot of memory fetching operations, lesser no of thread could perform the required operations in given time.

2.4 Validations and speed up

A similar code was developed on 16 Cores CPU machine to validate the results achieved using CUDA.

For each output value of CPU generated output and GPU generated output was compared and mean difference of the values were calculated. The calculated mean difference is 9x 10^-5 which is acceptable for this exercise. Following is the output generated using CPU as well GPU codes.

Table 1: Launch kernel initial timings and improved timings for each module:

<table>
<thead>
<tr>
<th>Module Name</th>
<th>Module Description</th>
<th>Initial Timing</th>
<th>Final Timing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sub</td>
<td>To convert the 8054 samples to 8K data and conversion in to float</td>
<td>126.04 ms</td>
<td>73.04 ms</td>
</tr>
<tr>
<td>CUFFT on Data</td>
<td>Calculate FFT of 18K x 8K data</td>
<td>40.31 ms</td>
<td>40.31 ms</td>
</tr>
<tr>
<td>CUFFT on replica data</td>
<td>Calculate FFT of 18K x 8K replica data</td>
<td>25.4 us</td>
<td>25.4 us</td>
</tr>
<tr>
<td>Muti</td>
<td>Multiplication of FFT’s of data and replica</td>
<td>116.475 ms</td>
<td>63.5 ms</td>
</tr>
<tr>
<td>ICUFFT</td>
<td>Inverse FFT of multiplication data</td>
<td>38.85 ms</td>
<td>38.85 ms</td>
</tr>
<tr>
<td>Scale</td>
<td>Scaling operation to take the square root of the data</td>
<td>142.305 ms</td>
<td>55.17 ms</td>
</tr>
</tbody>
</table>

Fig. 2 : Results using 16 cores CPU Machine
We got a significant **four times** improvement in range compression algorithm on GP–GPUs as compared to 16 cores CPU machine.

3. **Problem 2: Optimization of computation fluid dynamics code**

   » A Fortran code related to fluid dynamics algorithm at department of Fluid mechanics & thermal analysis division, Vikram Sarabhai Space Centre (VSSC) is getting executed on a machine having configuration windows 64-bit machine with 64 number of cores, clock speed of 2.4 Ghz & PGI compiler.

   » The best timings achieved is 965 sec using 1 core and 256 sec using multiple cores for 100 iterations and 24950 sec using 6-64 cores for 10,000 iterations.

   » The performance was not getting improved beyond 6-12 cores. There was a need to improve the execution time without affecting the precision of results.

   » Timings obtained at VSSC was taken as reference timing for comparing speed up.

3.1 **Approach & tools used for Optimization at SAC:**

It was observed that branching used in the algorithm is restricting degree of parallelism. Following techniques were used to optimize code. All the techniques were implemented using openMP constructs.

   i. Loop parallelization
   ii. Core binding
   iii. Locality of reference
   iv. Optimization compiler flags (prefetching, inter process optimization)
   v. Vectorization

At SAC, Code was executed on two different machine, A machine with similar clock speed to VSSC machine with more no of cores, and another machine with better clock speed & less no of cores. Machine configuration & compiler used at SAC are listed below:

   i. CentOS 7 Linux machine six cpu 24 cores machine with a clock speed of 2.5 Ghz
   ii. Intel Fortran compiler

3.2 **Results and timings obtained**

Timings and speed up obtained by increasing no of cores at SAC and w.r.t. speed up obtained at VSSC for 100 iterations and 10000 iterations by varying no of threads are tabulated in table 2,3,4 and 5.

**Table 2 : Timing on dual CPU 10 cores machine with clock speed of 3.1 Ghz for 100 iterations**

<table>
<thead>
<tr>
<th>Thread count</th>
<th>Timing</th>
<th>Speed up obtained by increasing no of cores at SAC</th>
<th>Speed up with respect to timing obtained at VSSC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>400 s</td>
<td>1x</td>
<td>2.41x</td>
</tr>
<tr>
<td>4</td>
<td>164s</td>
<td>2.43x</td>
<td>------</td>
</tr>
<tr>
<td>6</td>
<td>132s</td>
<td>3.03x</td>
<td>1.93x</td>
</tr>
<tr>
<td>8</td>
<td>98s</td>
<td>4.08x</td>
<td>2.61x</td>
</tr>
<tr>
<td>12</td>
<td>77.5s</td>
<td>5.16x</td>
<td>3.3x</td>
</tr>
<tr>
<td>16</td>
<td>64.5s</td>
<td>6.2x</td>
<td>3.96x</td>
</tr>
<tr>
<td>20</td>
<td>59s</td>
<td>6.7x</td>
<td>4.33x</td>
</tr>
</tbody>
</table>

**Table 3: Timing on 6 CPU 32 cores machine with clock speed of 2.5 Ghz for 100 iterations**

<table>
<thead>
<tr>
<th>Thread count</th>
<th>Timing</th>
<th>Speed up obtained by increasing no of cores at SAC</th>
<th>Speed up with respect to timing obtained at VSSC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>420s</td>
<td>1x</td>
<td>2.29x</td>
</tr>
<tr>
<td>6</td>
<td>156s</td>
<td>2.69x</td>
<td>1.64x</td>
</tr>
<tr>
<td>8</td>
<td>128s</td>
<td>3.28x</td>
<td>2x</td>
</tr>
<tr>
<td>12</td>
<td>103s</td>
<td>4.07x</td>
<td>2.48x</td>
</tr>
<tr>
<td>16</td>
<td>84.8s</td>
<td>4.95x</td>
<td>3.01x</td>
</tr>
<tr>
<td>20</td>
<td>77.4s</td>
<td>5.42x</td>
<td>3.30x</td>
</tr>
<tr>
<td>32</td>
<td>66.7s</td>
<td>6.29x</td>
<td>3.83x</td>
</tr>
<tr>
<td>64</td>
<td>51.0s</td>
<td>8.23x</td>
<td>5.01x</td>
</tr>
<tr>
<td>128</td>
<td>43.4s</td>
<td>9.6x</td>
<td>5.9x</td>
</tr>
<tr>
<td>192</td>
<td>37s</td>
<td>11.4x</td>
<td>6.9x</td>
</tr>
</tbody>
</table>

**Table 4: Timing on dual CPU 10 cores machine with clock speed of 3.1 Ghz for 10,000 iterations**

<table>
<thead>
<tr>
<th>Thread count</th>
<th>Timing</th>
<th>Speed up</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>13576</td>
<td>1.83x</td>
</tr>
<tr>
<td>6</td>
<td>13576</td>
<td>1.83x</td>
</tr>
<tr>
<td>16</td>
<td>6689s</td>
<td>3.73x</td>
</tr>
<tr>
<td>20</td>
<td>6182s</td>
<td>4.03x</td>
</tr>
</tbody>
</table>
Software optimization and parallelization approaches for solving scientific problems in image & data processing

Table 5: Timing on dual CPU 10 cores machine with clock speed of 2.5 Ghz for 10,000 iterations

<table>
<thead>
<tr>
<th>Thread count</th>
<th>Timing</th>
<th>Speed up</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed up with respect to timing obtained at VSSC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>15645s</td>
<td>1.59x</td>
</tr>
<tr>
<td>16</td>
<td>8465s</td>
<td>2.94x</td>
</tr>
<tr>
<td>32</td>
<td>6611s</td>
<td>3.77x</td>
</tr>
<tr>
<td>64</td>
<td>4931s</td>
<td>5.05x</td>
</tr>
<tr>
<td>128</td>
<td>4301s</td>
<td>5.8x</td>
</tr>
<tr>
<td>192</td>
<td>3742s</td>
<td>6.66x</td>
</tr>
</tbody>
</table>

We could achieve an optimization at single core level as well as a speed up by increasing no of cores without losing precision. As compared to timings achieved at VSSC on single core level, we could achieve nearly double the speed. We are able to scale this application with increasing no of cores. So compared to maximum speed up obtained at VSSC with 6-64 cores, we could get more than 6.5 times speed up with increasing no of cores. By further reducing branch conditions, we could get results in 9-10 hours which were taking 7-8 days of time without losing precision.

4. Problem 3: Computation of PI using High performance computing

There are various approaches used to compute the value of PI like polygon approximation area, infinite series and iterative algorithms [6]. We have worked with one researcher to compute value of PI using following approach.

![Figure 2: Approach used for computation of PI](image)

In this approach, circle is divided in to four quadrants and area of each quadrant is calculated. But to calculate area of quadrant with precision, each quadrant is further divided in to 10 segments. Further each bigger segment is divided in to number of small segments of the order of 108, 109 ... 1016. Area of smaller segments were calculated and added to form the area of bigger segments and then further added to form the area of the quadrant. Finally, area of the quadrant is multiplied with 4 to obtain the value of PI.

**Approach to calculate area in a small segment:**

Radius (r) is constant, x1 is known as segment size, and hence y1 is calculated using Pythagoras principle. Now, if we know x1 and y1, we can compute the area of a segment.

Earlier, Researcher had used 64-bit desktop machine & a serial version of the code to carry out this work. To process only one segment of a quadrant, for radius of 10^10, the desktop took 70 minutes of time. Similarly, for higher order radius of order 10^11, 10^12 & 10^13 it required days to do computations for a segment. Hence, it was not possible to carry out computation of pi with higher order radius using a desktop computer. So, for faster operations & better precision we have implemented parallel version of PI code on better architecture at SAC.

4.1 Implementation on HPC

We have been provided a serial code to execute on HPC with 144 cores and 700GB of memory. The course of actions to speed up the task is given below:

i. Serial code was compiled and executed on HPC machine for the purpose of verification of the results. On understanding the code and verification of the results, efforts for parallelization were put in.

ii. Looking at the problem, we have decided to break the independent segments and carried out processing of major segments on a separate core. And a generalized code was developed based on segment level parallelism as shown in figure 2.

iii. The code was then executed firstly on smaller segment size and further segment size was increased.

iv. As a result, all the segments of the order of 10^10 on 6 cores could be processed in seconds and all the segments of the order 10^11, 10^12 were processed in minutes using 60 cores. Further, all the segments of the order 10^13 which may take 400 days on 2.5 Ghz desktop machine, were processed in 8 hours on HPC using 100 cores.

![Fig. 3: Segment level Parallelization](image)
4.2 Validation Results of PI

Table 6: Value of PI computed using HPC and difference from textbook accepted value

(Results on HPC with radius 10^3)

<table>
<thead>
<tr>
<th>Value of Sum</th>
<th>Segment no</th>
</tr>
</thead>
<tbody>
<tr>
<td>081750564.68184846953101256773312</td>
<td>80-100</td>
</tr>
<tr>
<td>141897057.945229427750768695464</td>
<td>60-80</td>
</tr>
<tr>
<td>561750564.68184846953101256773312</td>
<td>00-60</td>
</tr>
<tr>
<td>785398187.308193668137921009306</td>
<td>00-100</td>
</tr>
<tr>
<td>3.141592749.2356793467251168403722</td>
<td>nc (Single quadrant)</td>
</tr>
<tr>
<td>3.141592749256793467251168403722</td>
<td>nc</td>
</tr>
<tr>
<td>3.141592535897932384626434</td>
<td>πT (Four quadrant)</td>
</tr>
<tr>
<td>0.000000005645861082624734403722</td>
<td>Diff</td>
</tr>
<tr>
<td>9.56458610826247343403722</td>
<td>10-8</td>
</tr>
</tbody>
</table>

In this exercise, quadrant was divided into 100 big segments and further smaller segments. Area of all the segments were calculated by approach suggested above. In table, area of 00-60, 60-80 and further 80-100 segments were added to form area of quadrant. Further, area of quadrant is multiplied by 4 to get the value of PI which is 3.1415927492356793467251168403722. This value is better than existing value of PI as per literature at that time with difference of 9.56458861082624734403722 x 10^-4. With HPC, we can obtain a significant improvement in term of performance as well as precision. The computations which may take 400 days of operations time on DESKTOP (2.5 Ghz) machine were performed in 8 hours of time on HPC machine with 100 cores (each core 1.86 Ghz).

5. Conclusions

With increasing data rate and size, it is mandatory to adopt parallel programming for all the software developers across the world to solve the problems in timely manner. Based on problem type, designers have to choose problem scalability aspects, target architecture, programming paradigms while maintaining the accuracy and precision of the results. In this paper, we have shared our fruitful experience of working on three different problems of parallelization and achieved results in terms of speed up gained. We could conclude that using computer parallelization and optimization we could solve scientific problems with better timings which in terms helps in better decision making.

6. Acknowledgements

The authors are thankful to then Chairman, ISRO Shri A. S. Kiran Kumar for providing the opportunity to work on problem of PI. The authors express sincere gratitude to Shri. D. K. Das, Director, Space Applications Centre for permitting the publication of this paper at the, CSI Journal of Computing. The authors are also grateful to Shri. N. M. Desai, Associate Director, Space Applications Centre for encouragement. Authors also thank then DD, SIPA, Santanu Chowdhury for his encouragements. Authors are also thankful to Dr. T. Jayachandaran, VSSC for involving us for optimization of computation fluid dynamics problem and Shri Hiralal Vankar for sharing his problem and experience on computation of PI. Suggestions from internal referees to improve an earlier version of this paper are sincerely acknowledged.

References

5] Thomas Bradley, Presentation on “Advanced CUDA Optimization Execution”, NVIDIA Corporation, Feb 2010

About the Authors

Mr. Jai G. Singla joined Space Application Centre (ISRO) in 2007 after completing B.E. in Computer Science and Engineering. He has worked in the field of high performance computing, 3D- visualization, GIS, photogrammetry, machine learning, virtual reality and software engineering. He has also contributed in development of software for data processing for science mission. His interests are development of new techniques, applications and software mainly using Indian remote sensing satellite data & products.

Mr. Kirti Padia joined Space Applications centre (ISRO) in 1983. He has worked for development of data processing software for ISRO Airborne and Spaceborne optical and microwave sensors like AHYSI, IRS, MSMR, O2SCAT, RISAT-1, MECHA-TROPIQUES as well as M’S and ERS-1&2. He has developed algorithms and software for In-SAR data processing using ERS tandem mission data. He has also significantly contributed in parallelization of algorithms and software. His interests are algorithms and software development for data processing related work for airborne and spaceborne Hyper Spectral sensors, microwave radiometers, scatterometers, SARs, altimeters, radio occultation payloads and 3-D visualization.
Image based Rain Removal Process for Vision Enhancement

Mrs. Bindhu. A. and Mrs. Sreeja S. S.
Department of Computer Science And Engineering, Marthandam College of Engineering And Technology, Kuttakuzhi

Video based rain or fog removal process is a challenging issue and it is studied extensively. The varying environmental conditions such as rainfall, snow and fog will affect the image features. These conditions degrade the effectiveness of the different processes such as feature extraction, image registration, object tracking and image restoration etc.,. The proposed system is a image based rain streak process to obtain the vision enhancement. The rain streak have two characteristics. They are, It have the repetitive local structure which is scattered over different directions of the video and the chromatic properties of rain streak. We use a MCA (Morphological Component Analysis) algorithm for the detection of rain droplets and other image parts. The photometric properties can be detached from the input to produce the vision enhanced image.

Keywords: Morphological component analysis, rain removal, vision enhancement

I. Overview

Machine visualization technologies are practiced for various principles such as observation, tracking in addition to intelligent transport systems (ITS) etc.,. These vision technological systems were implemented using various kind of algorithms such as feature extraction, feature matching, segmentation, classification etc.,. These algorithm’s effectiveness might be affected by the varying weather conditions and illumination variation. To improve the robustness of the system we have model a robust algorithm to perform the vision enhancement process. The dreadful climate conditions such as drizzle, haze and vapor will humiliate the recital of the outdoor vision system. The illustration of the rain component is as follows, Rain is the collection of water droplets which is comes at elevated speeds. Every rainstreaks can be represented as a sphere shaped mirrors which refracts and reproduce light appears on the camera, it produces a sharp intensity patterns in videos. The rain drops have the significant effect of spatial and temporal variations which humiliate the recital of the outdoor visualization systems. With respect to the above analysis, we have proposed a rain removal model that perform the decomposition morphological component analysis to eliminate/decrease the presence of drizzle effects without affecting the originality of the image/picture. This model proposes the image composition and various analysis steps to identify the presence of raindrops.

II. Background

2.1 Rain streak properties

2.1.1 Spatio-temporal Property

Rain streak is scattered randomly and falls to the earth at high speed. The spatio property states the rain streak features and the temporal property states the changing pixel values of the rain streak with respect to time enclosed by drizzle throughout the complete image.
2.1.2 Color Value Property

A rain streak is similar to a spherical lens; it becomes brighter when the light falls on it. The light energy penetrates through the rain drop, the luminance value is increased due to the reflective property which makes the rain drop clearer than the environment. The increase in chrominance values is proportional to the background. The chromatic property takes the Red, Green, and Blue values of the rain streak into consideration to calculate the chromatic property.

2.1.3 Photometric constraint

The substantial characteristics of the rain streak is described as photometric constraint. There are three factors to determine the characteristics of raindrop. Firstly, the brightness of the rain streak, then the environment scene radiances and finally the camera resolution. It also assume that rain streak have almost the same size and velocity.

III. Related works

Moreover, many image processing applications heavily depend on the mining of gradient information or directional information. It means the pixel intensity in x and y axis. The widely used feature extraction techniques that are dependent on the calculation of picture gradients they are recognition of features techniques, shape or edge detection approaches, and image properties descriptions algorithms. The effectiveness of these picture gradients properties mining techniques, can be considerably reduced by drizzle visible in the image. The dynamic properties are introduced by the drizzle with respect to varying times in the same direction. The correlation model [2] which extracts the properties of drizzle and a correlation model that describes the rain image properties. Visual attention techniques obtain the similarity map of image features programming for similarity at every local region on the image. A saliency map is an image that depicts each pixel's unique quality. The output produced by saliency map is the set of contours extracted from the image. The efficiency of the rain streak removal process is ruined if rain drops intermingled with the image region. A no of drizzle identification and elimination approaches based on adjusting parameters of the camera which is not applicable to the normal end-user cameras [6] and cannot be applied to existing captured picture/recorded data. To remove the drizzle from frames captured by a motion video capturing unit, the recital of the frames processing may be considerably decreased. Because the problem is the frame processing techniques typically execute the detection of drizzle , which is done by the interpolation of identified image features. These interpolated pixels are exaggerated by drizzles in every image sequence of dynamic environment because of camera movements and false motion assessment due to the influence of drizzle may reduce the recital of drizzle identification and removal process. The movement analyzing approaches can be implemented to obtain the properties of the camera motions, its recital can also be decreased by drizzles or huge movement activity. In the concept of stable properties of drizzle, it is difficult to identify the presence of these properties or identify the stable details from adjacent image sequences for reconstruction.

Proposed system

In the proposed system, the input image is first reprocessed for the quality improvement and decayed into the smoothened image parts (LF) and sharpened image part by the bilateral filtering process. The sharpened image details are decayed into “rain component” and “nonrain component” by obtaining the morphological component analysis process. The contributions are, i) Preserving the details of the video ii) decomposition of video to separate the video detail iii) automated process of rain streak removal.

Fig. 2: Block Diagram
IV. Contribution

The entitled system implements a vision enhancement process which is done by the rain streak removal process. In this system the decomposition process is applied to the input image which divides the picture into the smooth and sharp image regions. The decomposition is done by the bilateral filter, the low frequency part are the fundamental details of the image which will be maintained and the rain droplets and remaining edge information’s are considered as the high frequency components. Then, the Morphological component analysis is done on the high frequency components to identify the occurrence of drizzle and the other non-rain components. The dictionary leaning process is performed after the bilateral filtering process, the trained data is captured from the sharpened image part. Histogram gradient feature based dictionary learning method is used to divide HF part into two subdictionaries. The sparse coding is executed to perform the image disintegration process of MCA, in this the texture information in the HF component is isolated and it can be combined with the smooth part of the picture to perform the drizzle rectification process. The following section gives brief details of each each processes:

A. Preprocessing

In the preprocessing stage the input rain image is applied with the bilateral filter which preserves edge information of the input image. The bilateral filter is also called as smoothing filter and it act as a noise-removal filter. The output of this filter is the Low frequency such as basic information of the image and sharp edge image components, such as drizzle and the image related information.

Step-2: Patch extraction & Dictionary learning

In the patch extraction step, the the overlapping patches are extracted from the HF image parts.

The rain streak and texture information of HF part is divided into two learned dictionaries called as rain dictionaries and geometric sub-dictionaries. The image gradient is considered as the necessary feature for a rain component. From the rain HF part the feature descriptor obtains the rain component. The HOG descriptor separates the dictionary into rain dictionary and geometric dictionary of HF.

Fig. 3: Patch extraction

Fig. 4: Dictionary learning

Step-3: Sparse coding

Sparse coding is an approach to find the parse representation for a signal in image processing. Sparse coding is applied on the above two sub-dictionaries such as rain and texture to identify the sparse coefficients of the sharpened part of the picture. The sparse coding process results by separating rain component and image parts. Then the image part’s are combined with the smooth part to produce the rain removed picture.

IV. Algorithm

1. Input : Rain / Drizzle image.
2. Smoothening process: Smoothening filter process to produce Low frequency image elements and High frequency image part.
3. Construct Image Patches: Obtain the image patches from High freq components. 4. Perform sparse signaling representation and for each atoms extract HOG feature.
4. Use clustering Algorithm k-means to classify the extracted features into two clusters.
5. Based on the classification result the clusters are determined as rain dictionary and non rain dictionary.
6. OMP (Orthogonal Matching Pursuit) is applied for each patch in the rain dictionary.
7. Reconstruct the non rain components.
8. Combine the non rain component with the LF image components.
9. Output : Image after removal of rain streaks

6. Conclusion And Future Work

The proposed system is an effective method for rain streak removal. This method includes an automatic decomposition technique called as MCA for the rain streak removal process. The patches are constructed and the dictionary is learned for the decomposing process of rain streaks from an image. The rain component can be detached from the video to produce the vision enhanced video.
This system can be further enhanced by the implementation of CNN model for the effective removal of rain streaks. The dictionary learning process takes much time to compute the decomposition of drizzle and non-drizzle elements. So in the future system the learning process can be done in fast manner by the implementation of hidden layers.

References

About the Authors

Mrs. Bindhu A., currently working as a Assistant professor department of CSE in Marthandam Collge of Engineering And Technology, Marthandam. bindhu@macet.edu.in

Mrs. Sreeja S. S., currently working as a Assistant professor department of CSE in Marthandam Collge of Engineering And Technology, Marthandam. sreeja@macet.edu.in
Prediction of Long Duration Complex Human Activity in Videos

Ashwini S Gavali¹, S. N. Kakarwal² and D. V. Kurmude³

¹ Asst. Professor, Dept. of CS and Engg., Chh. Shahu College of Engg., Aurangabad. Email: ashugavl22@gmail.com
² Professor, Dept. of CS and Engg., P.E.S. College of Engineering, Aurangabad. Email: s_kakarwal@yahoo.com
³ Associate Professor, Physics department, Milind College of Science, Aurangabad. Email: dvkurmude@yahoo.com

Human activity Prediction from video is challenging task and has large demand in time critical application such as spotting criminal activity in surveillance system, healthcare assistance, video indexing and retrieval etc. This paper explores the use of sequential representation of human actions for predicting long duration human activity in video. This paper presents the novel framework for prediction of human activity by considering the sequential nature of human activity. This work is based on the fact that human activity is composed of constituent simple actions. So here we focused on real-time recognition of each constituent atomic action and predicted a class label for these local actions and then these local prediction results are used to predict the global class label for entire human activity.

1. Introduction

Lot of work has been done for recognizing Human action in short duration video and also has achieved significance performance. Human activity recognition is the task of automatically analyzing the activity in video. Activity recognition recognizes the activity after it is fully observed while activity prediction aims to recognize the activity before it is completely executed. This paper presents a novel approach for complex and long duration human activity prediction.

The ability to predict ongoing activity is challenging task because decision has to be made before the complete execution of activity based on available incomplete video. Also there are lots of challenges in activity prediction because of dynamic nature of video. Some of these challenges are as follow [1]:

- **Viewpoint:** Different location of camera may represent the same activity in different view such as top view, left view etc.
- **Background:** Clutter background often needs to extract human from background. This process is called as segmentation. The accuracy of object segmentation is highly depends on whether camera is static or moving.
- **Motion Style:** Same activity can be performed differently by different person and it can affect performance of system.
- **Speed:** Speed for performing any activity may be different for different persons.
- **Duration:** If the video is long, then it would generate large number of features which can increase response time of prediction system.
- **Occlusion:** If the human or some part of the video is occluded then it may affect performance accuracy of system.

The basic difference between action recognition and action prediction is that action recognition assigns a class label after the video is fully observed whereas action prediction predicts a class label based on the partially observed video. Hence the performance of action prediction highly depends on discriminative power beginning temporal segments of partially observed video.

In this paper we present an early prediction framework for long duration human activity. This work is based on the fact that activity is composed of multiple sequential actions. For example cooking activity consists of various actions such as taking vegetable, washing vegetable, cutting vegetable, etc. Here a database trained with small atomic actions is used to predict a class of long duration complex activity consisting of various sequential actions. We consider activity prediction as finding maximum aposteriori probability of class based on these sequential actions. Here rather than predicting a global class label for partially observed activity, we segment partial observed video into multiple atomic segment (we called them as Actionlet) and then we predict local class label for each atomic action segment in observed video. Finally based on local class labels, possible global class is predicted using maximum aposteriori probability. For example the activity preparing a salad is made up of local action units such as taking vegetables from freeze, washing vegetables, cutting vegetable etc. Now suppose we are just given incomplete video with two atomic actions as taking vegetables from freeze, washing vegetables then after predicting local class label for these two actions, we apply sequential prediction to predict global action class i.e. making a salad.
Here we choose to use sequential prediction for two reasons: First visual similarity may appear in action sequence. For example, consider the activities ‘preparing a sandwich’ and ‘making a salad’ have visual similarity in early stage. So if we predict activity at different progress level we can be able to distinguish visual similarity.

Secondly it is not always the case that human performs same activity for long time. Say for example consider a situation, a human enters in the room he first go to kitchen and starts cooking but later on he stops cooking and starts reading recipe book. In this case if global prediction approaches may not be able to distinguish these two different classes of activity. But if we use sequential prediction for each atomic segment related to cooking and reading then some segment will be predicted as cooking and some segment will be predicted as reading and hence it also indicates human is performing two types of actions, and hence it improves the real time performance of the system.

2. Literature Review

As human action recognition has numerous applications in variety of domain, Human action recognition from pre-recorded videos has taken a large attention from last two decades from last few years and lots of work has been done. Some of recent approaches for action recognition are as in Qiuxia et al. [2], Manoj et al. [1], Liu et al. [3] etc. And most of these approaches also have reported significance performance.

The success of any intelligent system is based on how fast it responds to real world situation. Some of the researches have focused on developing such intelligent system where human activity can be inferred before it is fully observed.

Cao et al. [4] used a sparse coding based method for predicting human activity from partial video. They first divided each activity into multiple temporal segments and then combined the likelihoods at each segment to achieve a global posterior for the activity.

Ryoo et al. [5] addressed early activity prediction problem by modeling activity using bag-of-words paradigm. They used 3D XYT spatio-temporal features to model activity and then vector quantized these features using bag of word paradigm to form class specific visual words. Dynamic programming algorithm is used to dynamically classify observed video at different observation level. This approach is well suited to handle the sequential nature of human activities and to model noisy observation.

The above approaches are suited only for short duration simple activity. They not suitable for predicting activities with hierarchical structure or for long duration complex activity consist of number of constituent action. Another limitation of these approaches is that they predicts the activity at different progress level by assuming that the duration of the activity of same class is same every time. But this is not the fact as action video of the same class may be variation in time, pose, appearance etc.

Kang Lai et al. [6] addressed the problem of long duration complex activity prediction by mining temporal sequence patterns. They worked on two action prediction scenarios: action only prediction and context aware prediction. This work detects the different atomic segment using motion velocity peaks. They used probability suffix tree (PST) to model the relationship between constituent actions and Sequential Pattern Mining (SPM) to predict next possible action as sequential prediction problem. This approach is suitable for activities with hierarchical structure or repetitive structure, but it is not suitable for the activities with shallow structures. Also this approach is based on the assumption that activity is always performed in same sequence, but in real world the sequence of sub-activity may change.

Wenbin et al. [7] used recurrent neural network to solve activity recognition as sequence prediction problem. They represented each activity over hierarchical label space. This approach first predicts activity at course level and then uses these results for fine grained recognition.

Tianlan et al. [8] propose hierarchical moves to describe human movements at different level of granularities. This work used max margin framework for activity prediction.

Yu kong et al. [9] introduced a framework for predicting activity at different progress level i.e. local and global progress level. This approach captures relationship between different progress level and partial observation to predict global class label for incomplete action.

Main Contribution of proposed long duration activity prediction framework:

All above three approaches predicts the long duration human activity at different progress level. But most of them have used fully observed long duration video for training and then testing video is compared with training video at different progress level assuming that the activity of same class is performed in same sequence and with same duration every time. Such approach is not well suited for outliers where there is variation in activity duration, pose and appearance etc. Also visual similarity may occurs in different action class, therefore rather than training same visually similar segment for every class, the better way is to train them only once for all classes.

Also these approaches are not suitable for real world action prediction, as most of them used dense optical flow based dense trajectory features [20][21]. As dense optical flow is computationally expensive this limits the work to scale in real world. To overcome this, we used mpeg flow based video descriptor [22] for feature extraction from videos.

Our approach differs from the previous approaches as it uses the database trained with every single action is used to predict the sequential activity consisting of multiple constituent’s actions. Here the classification task is not performed by matching progress levels and hence it is suitable for outlier cases where there is variation in activity duration, action sequence, pose and appearance etc. belonging to same class. For example consider the activity preparing sandwich. It consists of constituents actions such as taking vegetables from freeze; wash the vegetables, cut the vegetables etc. Here
rather than training database with fully observed activity of sandwich preparation as like previous approaches, we learned dictionary for every possible single atomic action such as taking vegetables from freeze, wash the vegetables, cut the vegetables etc. This approach minimizes the time for training visually similar segment for every class. Now suppose we have partially observed video for testing with only two actions taking vegetable and cutting vegetable. Then our approach first segment the video into multiple atomic segment and find the class label for these segments and based on this current observation, it predict the future activity as sequence prediction problem.

Rather than common bag of word approaches, our approach used sparse dictionary learning algorithm for activity encoding.

3. Proposed Approach

Training:

Our method takes all possible small atomic action units (Actionlet) such as washing vegetable, cutting vegetable etc. for training and models the relationship between partially observed video and trained actionlet at testing phase. Main steps in model learning are as follows:

i) Feature Detection and Extraction: For each single atomic action unit we first extracted dense Histogram of Optical Flow (HOF), Histogram of Gradient (HOG), Motion Boundary Histogram (MBH) features [22].

ii) Dimensionality reduction: These densely extracted descriptors are high dimensional. So we use Random projection for reducing the dimensionality of feature vector [12].

iii) Dictionary learning: Instead of common bag of word approach, we use sparse dictionary learning. Given a set of input vector for each training video, the over complete dictionary is leaned and corresponding sparse representation is obtained using densely extracted features [13]. Class specific dictionaries are learned by solving sparse approximation problem using K-SVD algorithm [14].

Testing:

i) Temporal segmentation: Given a partial video consisting of complex long duration activity, first step is to temporally segment the given video such that each segment of video consists of meaningful atomic action. This step is carried out using Superframe segmentation proposed in [11]. The idea is to find the boundaries in video where significant changes in motion occur, and then cut the video accordingly into multiple segments.

ii) Feature Detection and Extraction: For each single atomic action unit in testing video extract dense HOG, HOF and MBH features.

iii) Classification: The class of each observed segment is recognized using learned dictionary and Random Sample Reconstruction (RSR)[13].

iv) Dynamic Prediction: Using the class label for each local segment of observed video, predict the global class label for unobserved video by computing Maximum A posteriory probability [15].

Fig. 1: Overall Pipeline of Proposed Approach.
3.1 Temporal Segmentation

Temporal segmentation is the first step of our activity prediction model. Given a partially observed video, the goal is to segment video into multiple segment by finding atomic actions in video (we called this atomic action as actionlet) where significant action changes occurs. As video consists of scene and each scene is represented as sequence of frames, the goal of actionlet detection is to represents observed video as sequences of atomic segment which represents discrete action units. In short goal is to find action boundaries based on the frame indices of video where significant action changes takes place and segment the video accordingly. For example if person is changing action from walking to seating then these two different actions should be segmented. We used superframe segmentation [11] for finding actionlet. This work used KLT tracker for tracking points in video and then used motion magnitude to segment the video into superframe (cuttable segments).

3.2 Action Representation and Feature Extraction

There are several local feature based approaches for video representation. Keypoint based approaches such as Space Time Interest Point (STIP) has been popularly used in most of the previous approaches. But the limitation of keypoint based approaches is that if the scale of the moving object is small, then it generates very few keypoints which are not enough to

Discriminate activity. We used dense HOG, HOF and MBH for extracting spatio-temporal features across the frames in videos.

The Dense HOF/HOF/MBH proposed in [10] is computationally fast and also generate large number of features. These features capture the spatial as well as temporal information of videos.

HOG descriptor captures the static appearance information of the moving object in video while HOF and MBH descriptor captures the motion information. HOG uses the orientation of image gradients while HOF and MBH descriptors are based on orientation of optical flow vectors. HOF simply quantizes the orientation of optical flow vectors while MBH first splits the optical flow into horizontal direction and vertical direction, and then quantizes the derivatives with respect both direction.

3.3 Dictionary Learning and Dimensionality Reduction

Unlike previous approaches that used bag of word modeling, we used sparse dictionary based representation which is generalization of bag of word modeling.

Consider a set of descriptors extracted from video using dense HOG/HOF/MBH, where every descriptor is of length 1. Then feature matrix can be represented as $D \in \mathbb{R}^{n \times m}$. These densely extracted descriptors are high dimensional. So we used Random projection for reducing the dimensionality of feature vector. These high dimensional descriptors are projected onto n dimensional subspace (n<<d) as:

$$ Y = RD $$

Where R is random matrix $D \in \mathbb{R}^{n \times l}$. Here D is the reduced data matrix.

Next, for all training activity video, our approach first detects atomic actionlets and encodes each actionlet using sparse dictionary.

$$ D \in \mathbb{R}^{n \times m} (n < m) $$

Then overcomplete dictionary is constructed by concatenating all these features from training samples. Solving such over complete dictionary is very difficult as it creates undetermined system of linear equations $b=DX$ which can have infinite number of solution.

Consider lower dimensional descriptor

$$ Y = \{y_1, \ldots, y_p\} \subset \mathbb{R}^n. $$

The goal of dictionary learning is learn a dictionary $D \in \mathbb{R}^{n \times m}$ over which Y has sparse representation $X = \{x_1, \ldots, x_p\} \in \mathbb{R}^n$ such that it should not contain more than k nonzero elements.

Formally learning sparse dictionary is the optimization problem which can be written as:

$$ \min_{D, X} \{F(\hat{Y} - DX)\} \text{ subject to } \|x_k\|_0 \leq k, $$

Where $F(\hat{Y} - DX)$ represents the sum of residual matrix which is calculated as:

$$ F(\hat{Y} - DX) = \sum_{i=1}^{p} |y_i - \sum_{k=1}^{n} x_{ik}d_k| $$

In above equation while computing X, D is kept fixed. There are various algorithm to solve above equation such as Basis Pursuit(BP), Orthogonal Matching Pursuit(OMP) etc. Here we used OMP to solve all sparse approximation problems.

2. Dictionary update. In this step rank-one approximation of residual matrix is calculated as:

$$ E_i = Y - \hat{Y} = Y - DX_i. $$

Atoms of dictionary are then updated sequentially based on above error. Finally class specific dictionaries are learned for each training class by iteratively computing sparse coefficient and updating the dictionary.

3.4. Classification

Given a K dictionaries for K number of classes, our classification model computes reconstruction error using RSR algorithm.
Given a set descriptors extracted from query video segment $Q$, RSR algorithm computes the reconstruction error $e_i$ for each class as:

$$e_i = \left\| Q - D_i \hat{X}_{Q} \right\|^2_2,$$

Where $\hat{X}_{Q}$ is the sparse approximation of $Q$ over each $D_i$ obtained as:

$$\hat{X}_{Q} = \arg \min_{\hat{X} \in \mathbb{R}^m} \left\| Q - D_i X \right\|^2_F \text{ s.t. } \left\| X \right\|_0 \leq k,$$

Finally the class having minimum reconstruction error is chosen as the output class.

### 3.5 Activity Prediction Model

Our approach estimates the local action units for each segment of observed video and then these local action units are used to predict global class label for ongoing activity.

The aim is to predict the global activity class label $C$ for partially observed video $O[1:t]$ where $t$ is the progress level of partially observed video of length $T$.

Let $O$ is the unfinished video which is progressed up to length $t$. This activity prediction model segments the observed video by dynamically detecting actionlets $[O_1, O_2...OT]$ such that each actionlet represent atomic action. Then we extracted dense HOG HOF and MBH features from every segment and finds the actionlet label $L_i$ of each observed actionlet $O_i$ using sparse dictionary classifier. It means that every actionlet $O$ from observed video is mapped to some training Actionlet $A$ as

$$L_i \in A_i, i=1,2,...,N$$

Where $N$ is total number of trained actionlets from all classes. Now, the Database is created for each training activity class $C_j$ as a sequence of actionlet $[A1,A2,A3...AN]$, Here each class of activity is modeled by sequence of actionlets. For example in following table Class C1 is modeled by three actionlet sequence A1, A5 and A6.

<table>
<thead>
<tr>
<th>Activity Class</th>
<th>Actionlets</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>A2</td>
</tr>
<tr>
<td>C1</td>
<td>Yes</td>
</tr>
<tr>
<td>C2</td>
<td>No</td>
</tr>
<tr>
<td>C3</td>
<td>Yes</td>
</tr>
<tr>
<td>C4</td>
<td>No</td>
</tr>
</tbody>
</table>

This matching process is to be repeated for every actionlet/segment that is detected as video progress. The idea is take likelihood computed in previous observation and updates the likelihood for entire observation. Based on this incremental likelihood an optimum activity class is decided that best describes the observed video.

### Table 1: Representation of the Actionlet Sequences for Activity Class.

<table>
<thead>
<tr>
<th>Activity Class</th>
<th>No of Actionlet Matched</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>2</td>
</tr>
<tr>
<td>C2</td>
<td>3</td>
</tr>
<tr>
<td>C3</td>
<td>1</td>
</tr>
<tr>
<td>C4</td>
<td>2</td>
</tr>
</tbody>
</table>

Here optimum likelihood is for activity class C2 because actionlet label matching for class two are maximum. So observed video is more likely matching to activity C2.

### 4. Experimental Results

We have tested our approach on two dataset MHOI and MPPI cooking dataset.

#### Results on MHOI Dataset:

MHOI is the daily activity dataset such as “answering a phone call”, “drinking tea” etc. In each class of activity human
interacts with some object. These dataset activities are short duration activities consists of around 2 to 4 atomic actions (actionlet) such as grabbing the object, putting it back etc. There are total 6 different types of activities each of which is performed by 8 to 10 subject.

We segmented and trained all possible actionlet from training samples. We used 70 percent subjects from each category for training and remaining 30 percent for testing. We compared our approach with various existing approaches at different observation level. Our approach performs well compare to other approach in terms of accuracy as well as number of training samples used.

Results on MPPI Dataset:

MPPI dataset is the cooking activity dataset such as “making a salad”, “making a sandwich” etc.

In each class of activity human interacts with some object. These dataset activities are long duration complex activities consists of around 20 to 125 atomic actions (actionlet) such as cut slices, pour, or spice. There are total 65 different activities (actionlets) performed by various actors. There are total 14 different types of dishes each of which is performed by 3 or 4 subjects. Totally there are 44 videos of length approximately 8 hours.

Table 2: Comparison with State-of-The-Art on MHOI Dataset.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Training samples used</th>
<th>Observation Ration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>20%</td>
</tr>
<tr>
<td>Integral BOW[5]</td>
<td>Leave-One-Out</td>
<td>0.32</td>
</tr>
<tr>
<td>Dynamic BOW[5]</td>
<td>Leave-One-Out</td>
<td>0.40</td>
</tr>
<tr>
<td>BOW+SVM[17]</td>
<td>Leave-One-Out</td>
<td>0.30</td>
</tr>
<tr>
<td>HMM[6]</td>
<td>Leave-One-Out</td>
<td>0.23</td>
</tr>
<tr>
<td>Action Only Model[6]</td>
<td>Leave-One-Out</td>
<td>0.37</td>
</tr>
<tr>
<td>Our Approach</td>
<td>70 percent</td>
<td>0.35</td>
</tr>
</tbody>
</table>

Fig. 3: Sample Actionlets From MHOI Dataset

Fig. 4: Sample Actionlets from MPPI Dataset

Table 3: Comparison with State-of-The-Art on MPPI Dataset.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Training samples used</th>
<th>Observation Ration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>20%</td>
</tr>
<tr>
<td>HMM[6]</td>
<td>Leave-One-Out</td>
<td>0.49</td>
</tr>
<tr>
<td>Action Only Model[6]</td>
<td>Leave-One-Out</td>
<td>0.58</td>
</tr>
<tr>
<td>Our Approach</td>
<td>70 percent</td>
<td>0.52</td>
</tr>
</tbody>
</table>
Conclusion and Future scope

In this work we presented sparse dictionary learning and maximum Aposteriori probability based approach for human action prediction. This approach predicts the global action class while recognizing local action units in videos. This approach is well suited for predicting long duration activity consisting of various sub-activities.

In this work we have just focused on predicting single person activity. In future we will extend this approach for the scenario where multiple people are present in video.

References

17] Ivan Laptev, MarcinMarszalek, CordeliaSchmid, Benjamin Rozenfeld,”Learning realistic human actions from movies”, in Proc. CVPR08, Anchorage, US.

About the Authors

Ashwini S Gavali is presently working in Department of Computer Science & Engineering of CSMSS Chh. Shahu College of Engineering. She has Completed M.Tech in CSE from Dr. BAMU Aurangabad. Presently she is pursuing Ph.D from Dr. BAMU Aurangabad. She can be reached at ashugavali22@gmail.com.

Dr. S. N. Kakarwal [CSI-F8000602] is presently working as Professor in Department of Computer Science & Engineering of P.E.S. College of Engineering, Aurangabad, MS-India. Her research interests include Image Processing, Pattern Recognition. She has published 20 papers in International Journals, 33 papers in International Conference and 12 recognitions in National Conferences. She has published 11 articles in CSI Communications. She can be reached at s_kakarwal@yahoo.com.

Dr. D. V. Kurmude is presently working as Associate Professor and Head in Department of Physics at Milind College of Science, Aurangabad. MS-India. He is member of Computer Society of India. He received Bharat Vikas Award -2018. He has published more than 20 research papers in International Journals and conferences in the area of material science. He has published EIGHT articles in CSI Communications. Authored a book on Electrodynamics useful for Physics undergraduates. The book is published by international publisher. He can be reached at dvkurmude@yahoo.com.

CSI Journal of Computing | Vol. 3 • No. 3, 2017
A Hybrid Approach for Anatomical Retinal Structure Segmentation using Adaptive Fuzzy Thresholding

Syed Akhter Hussain¹ and Deshmukh R. R.²

¹ Aditya Engineering College, Beed (MS) India - 431122
² Dept. of Computer Science & IT, Dr. Babasaheb Ambedkar Marathwada University, Aurangabad (MS) India - 431 001

Eye examination plays a vital role to determine health concerns. Today retinal screening is essential routine checkup to find hidden health issues in pre diabetic and diabetic patient. For screening of retina ophthalmologist need segmented patterns of retina but it is very tedious and time consuming process to get accurate segmentation of vessels, segmentation of optic disc and anomalous lesions which may affect the successive clinical treatment, our paper presents automated system for segmentation and recognition of retinal lesions. Our hybrid system uses a powerful hybrid segmentation algorithm by combining the characteristics of both adaptive fuzzy thresholding and mathematical morphology. In our hybrid system we are using publically available data sets DRIVE and STARE for vessels, DRISHTI-GS is for optic disc and DIARETDB1 for exudates lesions. Using this we achieve competitive segmentation performance furthers it can deal with further various anatomical structures.

Key Words: Retinal examination, vessel segmentation, morphological operations, exudates segmentation, optic disc segmentation FCM, adaptive local thresholding, binarization, optic Nerve head.

I. Introduction

Retina resides next to the optic nerve it receive light through focusing optics and transfer it into neural signals send it to brain for further visual signals. The inherent changes in retinal physical structure play a critical role for retinal screening. For detecting retinal diseases resembling diabetic retinopathy (DR), Glaucoma Hyper tension, hypernyms (ROP), Age related Macular Degeneration (AMD) Morphological variations of retinal anatomical structure play a vital role in diagnosis. The major diesis is Diabetes which affects over all human body conditions the common reason of blindness is diabetic retinopathy which affects retina. Around half of blindness is control through early identification and retinal analysis. Early detection of diabetic retinopathy is done by retinal screening in diabetic retinopathy no early symptoms can be noted it arise progressively with respect to time.

In the starting phase of diabetic retinopathy it affects the blood vessels of retina this can be recognized by detection of microaneurymas diabetes damage optic nerve head (ONH) which change the shape of the OD it increases the intra ocular pressure due to which size of vessels are increased. Walls became porous of the retina it allows the leaking of aqueous humor that leads harden of the exudates. Earlier detection of retinopathy may help for treatment to getting verse else it can be complicated to treat [1].

Ophthalmologist perform analysis of retina using high end fundus cameras like scanning laser ophthalmoscope(SLO) ophthalmologist uses 2D retinal as well as segmented version of retina for pre diabetic and diabetic retinopathy, and several health issues can be avoid when it detects early [2]. These limitations of human observation motivated us to develop an automated system which gives more accurate retinal analysis without any need of synthesis and texture analysis. In this paper we present combination of both mathematical morphological theory and fuzzy sets using futures of them we get accurate segmented results. The universal framework of hybrid flexible soft thresholding system depicted as in Fig1.

![Fig. 1: Framework of segmentation system with output](image-url)
A Hybrid approach for Anatomical Retinal Structure Segmentation using Adaptive Fuzzy Thresholding

In our proposed system we subdivide our system in three functionality based subsystem for vessel, optic disc and exudates lesions. It is fully hybrid arrangement of mathematical morphology features and fuzzy set theory. mostly for image segmentation tasks in biomedical imaging systems thresholding is well known method [3]Thresholding gives us binary images in our system thresholding gives us a global value which is responsible for maximum partition between special classes in the image here we mean different tissues Thresholding segmentation is used where dissimilarities of tissues and organs are identify by dissimilar gray levels because objects in natural scenes are different then medical images. Here global thresholding implement a pixel-wise approach during soft transition in between different gray levels on the basis of fundus images it affects segmentation process because of irregular illumination or noise distortions. Same intensity level pixels segmented into same objects it is the drawback of global thresholding with particular rigid value to overcome this we adopt region based thresholding technique. we combine rule based and machine learning method. Hence adaptive fuzzy thresholding elaborate rigid segmentation phase and morphological operations elaborate the soft segmentation. In this paper we present a firm segmentation system which confine, recognize and extract several genetic anatomical structures which has extremely diverse features in median segmentation session with accurate segmentation.

II. Proposed System

Our work present a hybrid system which combines adaptive local fuzzy thresholding and mathematical morphology the flow chart is shown in Fig.2.

Pre and post processing are done by morphological operators and for processing phase adaptive fuzzy thresholding is responsible. Morphological operator works on binary images as well as structural image as an input and using complement set operator combine both. Our system is based on three major phase’s extraction of ROI, coarse segmentation and soft segmentation. In first stage we choose extract target region from the raw retina image for enhancement of accuracy with respect to retinal anatomical structures here we focus on vessels, exudates lesions or optic disc. For region of interest retinal images undergoes few pre-processing steps first at all morphological operations responsible for initial identification of the target region. Although this stage is an initial one but it has highly effective for very last segmentation accuracy of the fuzzy processing stage. The region of interest produce the input for local adaptive fuzzy thresholding, to get hard segmented image and for soft segmentation morphological operations are performed using binarization and convex-hull transform we obtained the final segmented images resembling vessels, optic Disc or exudates with respect to target retinal structure. The basic steps of our system are pictorially defined in Fig.3.

A. Phase I: Retinal Image pre-processing

The most important goal of this phase is region of interest extraction it selects the retinal structure of desired portion of retina to enhance the overall performance and reduces the computational cost. In the preprocessing step once we obtain extracted retinal images from the anatomical structure region on the basis of their properties and features we quit raw images. Here one thing is clear preprocessing steps may be different which are truly based on target anatomical structure. In the Following section we have explain pre-processing steps for each anatomical structures.

1) Region of interest of Retinal Vessels

This is our first phase of image pre-processing technique in this we focus on accurate vascular structure without observing macula and optic nerve head as we know retinal vessels are highly low contrast then fundus image surroundings. Here we observe that retinal vascular...
structure is non uniform with respect to size and contrast level because there are number of spread branches on retinal fundus images to overcome this problem F.Zana et al. [4] and Hengen et al. [5] proposed segmentation techniques in two stages. In first step morphological filter is applied to obtain linear structure for second stage they applied hysteresis thresholding to obtain binary vessel image. We want only linear structure following we mention mathematical formula to get ROI of retinal vessel structure.

\[ I^v_{ROI} = I^b_{retina} - I^a_{retina} \quad \ldots \quad (1) \]

Where

\[ I^a_{retina} = 3^{\text{imreconstruct}} (I^a_{retina}) \quad \ldots \quad (2) \]

And

\[ I^b_{retina} = \beta (I^\text{comp}_{retina}) = N_a \cdot I^\text{comp}_{retina} \cdot E \quad \ldots \quad (3) \]

Here \( I^b_{retina} \) is homogenous background without retinal vessel structure and \( I^a_{retina} \) is linear structure with vessel structure.

Optic nerve head is little blind spot its shape is slightly oval it contains optic cup it is a location where each and every nerve fibers congregate to form the establishment of the optic nerve [7]. The cup and disc margin are connected by neural retinal rim. We get these variables from raw retina through extraction of the optic disc region. The rounded oval shape of optic disc we use Hough transform for region of interest it extract the center of the neural retinal rim of the optic disc, and consequently the square window around the optic disc, representing the optic disc ROI for that we can take an account of the following steps:

a) Edge detection

Edge detection is used to find the boundaries using Hough transform in preprocessing phase it gives edge map of boundary pixels set which describe the border line of optic disc. Hough transform had capability to give accurate and efficient center of optic disc by employing perfect edge detection technique. To get this FCM clustering algorithm was applied. Before FCM algorithm following are the preprocessing steps for retinal images to get accurate edge diagram.

Step1: extraction of retinal red layer that is

\[ I^R_{retina} = 3^R (I^R_{retina}) \quad \ldots \quad (1) \]

Step2: enhancement of extracted retina layer

\[ I^e_{retina} = 3^{\text{CLAHE}} (I^R_{retina}) \quad \ldots \quad (2) \]

Step3: applying median filtering for further enhancement and then it apply to FCM algorithm as shown in Fig. 7
A Hybrid approach for Anatomical Retinal Structure Segmentation using Adaptive Fuzzy Thresholding

Step 4: FCM algorithm applied on enhanced image it aggregate optic disc in one cluster and 24 for other tissues.
Step 5: thresholding is applied for binarized version using

\[ I_{FCM}^{bw} = \begin{cases} 
1, & I_{FCM} = c \\
0, & Otherwise 
\end{cases} \]  

(3)

b) Hough transform

The Hough transform is used to detect parametric curve it uses gradient smoothing and voting parameter space to detect lines in this case we depict perpendicular lines to edge point of a circle which overlap in the center of the circle we get center of circle brighter. In our case we are having only one optic disc of each retinal image, hence we use. Parametric space voting method. Mathematical formula can explain better.

\[ (x-f)^2 + (y-g)^2 = r^2 \]  

(1)

Here we are using xy plane to depict optic disc. Here f and g are the coordinates and r is radius. Hough transformation is applied at edge pixels with respect to x,y intensity levels. To define circle edge detector consider points positioned on the curve as

\[ f = r \cdot \sin(\theta), \quad \theta \in [0, 2\pi] \]  

(2)

\[ g = r \cdot \cos(\theta), \quad \theta \in [0, 2\pi] \]  

(3)

Accumulator array is used for top fitting of edge as follows.

\[ A(\hat{r}, \hat{\theta}) = A(\hat{r}, \hat{\theta}) + 1 \]  

(4)

Hence we obtain one circle which match optic disc circle.

c) Optic disc region of interest square Window

Hough transform gives us an ideal circle with respect to a known radius r. By choosing this value our system create a square windows which equals 2r pixel width using corp. function of matlab ultimate I_{ROI} image. Extracted image shown in Fig. 8

Fig. 8: (a) unrefined retina image having exudates lesions. (b) Exudates region of interest.

3) Region of interest of Exudates Lesions

This phase is very important one of the most common diseases called retinopathy will be monitor with exudates regions fig.8.a clearly shows white or yellow soft abnormal regions having non uniform shapes and size. We are using same procedure to find extraction of ROI same as OD extraction. We replace OD as black region. Here our system can’t misclassified optic disc region in segmentation phase as shown in Fig.8.b

B. Phase II: Image Processing

In this phase we focus on segmentation technique which depends on fuzzy theory by means we choose local fuzzy thresholding. For our proposed system we merge mutually adaptive local thresholding and spatial local information based thresholding[8]. This stage consist following steps.

1) Stage I: Fuzzy Modeling

For fuzzy modeling we want fuzzy membership values which is purely based on membership function to get this we have to follow certain steps as mentioned bellow

a) fuzzification of an Image

It is a sorting of image coding where input can be viewed as undistributed fuzzy sets it group vessels as well as optic disc and remaining part as a background without no loss of characteristic of other anatomical structure.

b) Composition of Fuzzy sets

It is one type of preprocessing phase which gives region of interest it has several overlapping zones defining each and every of a fuzzy set which correspond to the target retinal structure such as exudates, vessel and optic disc, mainly zones the region of pixels those belongs to both fundus surroundings or retinal structure, mathematically it can be specified.

c) Composition Fuzzy relations

In this phase we want extraction of centroid from clusters it is used in membership function. Here we use mathematical representation of membership function using fuzzy algorithm [9].

2) Stage II: Aggregation of Fuzzy model

These stage spatial filters are applied on fuzzy planes for modification of fuzzy membership values. as a replacement for of spatial filter here we take values of corresponding members it can assign mathematically, for vessels and exudates lesions we use median filter and for optic disc we use linear filters because it contain large number of vessels and unidentified boundaries.

C. Phase III: Image Post-Processing

This is last step of our proposed system mainly it is segmentation stage. Following are the steps involved in this phase.

Step 1: the output of aggregation of fuzzy model follows binary thresholding with respect to target anatomical structure to get binirization. Hence binirization gives us set of pixels
which are misclassified, secluded and false pixel.

**Step2:** for cleaning of pixels we use morphological operators. It use target anatomical structure to compare the set of pixels detected previously. It help to remove unwanted pixels as shown in Fig.9.

**Step3:** for vessels it produce smooth region of edges.

**Step4:** for OD and exudates lesion morphological dilation operations are performed with convex hull is to obtain smooth version of OD and exudates lesions as shown in Fig.9 and Fig.10.

![Fig.9: post-processing of retinal vessel segmentation (a) output (b) binarization (c) morphological operations.](image)

![Fig.10: OD segmentation (a) output (b) binarization (c) morphological operations (d) convex hull transform.](image)

**III. Results & Discussion**

Our system enormously works on various anatomical structure having diverse features. For development of this system we seek advice from ophthalmologist most of the ophthalmologist give as valuable technique how they deal with retinal analysis one thing we conclude from our discussion eye diseases develop slowly it can only detect taking routine checkups when sudden health symptoms occurs it can’t detect in initial retinal analysis and it is tedious to analysis by human observer to overcome this we try to develop hybrid automated system for anatomical structures. In early stage of system development we have work on vessel segmentation and extraction we compare this technique with other research work we used fully connected random field model for segmentation it gives drastic results but in case of retinopathy where we unable to find bulgy patterns in vessel for some cases to overcome this problem we have done the work simultaneously on both optic disc and exudates lesion. Finally we merge all this technique in one hybrid system to better understand our systems functionality we estimate certain steps which mentioned as follows.

1. Input Image
2. Resize image to 256x256 resolution
3. Extract the RGB layers from input image
4. Use the Red layer image for further processing
5. Complement the image for vessel extraction
6. Apply adaptive histogram equalization to enhance the image.
7. Apply morphological operations to remove the disk part
8. Apply median filter and adjust the image for vessel extraction.
9. Convert the image to binary by evaluating the gray threshold.
10. Evaluate the boundaries and apply the segmentation legend to display the extracted vessels.
11. Use red layer for disc extraction, apply the histogram equalization for enhancement.
12. Apply FCM based clustering to extract the clusters to segment the disc part.

![Fig.10: Post-processing exudates lesions detection (a) output (b) binarization(c) morphological operations (d) convex-hull transform.](image)
A Hybrid approach for Anatomical Retinal Structure Segmentation using Adaptive Fuzzy Thresholding

IV. Conclusion

This paper presents a hybrid system for retinal vessels segmentation, optic disc segmentation and exudates lesions segmentation using fuzzy set features and morphological operations. From an ophthalmologist opinion first step is extraction of retinal structure. In computerized screening. The output of our proposed subsystem we integrate these three subsystems to acquire contained clinical information. As a research fellow our contribution is we provide a single hybrid system for detection of retinal anatomical structure. Hence no need to design system for separate system for each anatomical retinal structure. As per performance our system work under any type of biological anatomical structure it is highly robust it gives better accurate results than human observer we experienced our system performance on publically available data sets like DRIVE,CHASEDB1, STARE, DRITSHTI-GS, DiaRetDB1 and HRF. Experimental results show our results are superior with respect to specificity, sensitivity and accuracy. Finally we can say that we achieve powerful system that can be work smoothly when we combine morphological operations with fuzzy sets. We achieved replacement of rigid world with fuzzy one.

References

About the Authors

Syed Akhter Hussain received a Bachelor's degree in information technology and M.E. Computer science and Engineering from Dr. B.A.M. University Aurangabad (MS) India. He is pursuing a Ph.D. Degree in computer science and engineering from Dr. B.A.M. University Aurangabad. His research interests include image processing and biomedical image processing.

Deshmukh R. R. has received a degree in M.E. computer science and engineering from Dr. B.A.M. University Aurangabad (MS) India and Ph.D. in computer science and engineering from Dr. B.A.M. University Aurangabad (MS) India. His research interests include Human Computer Interface, Digital Speech Signal Processing, Image Processing and Pattern Recognition, Data Mining and Remote Sensing and Hyper-spectral Signal and Image Analysis. Human Computer Interface, Digital Speech Signal Processing, Image Processing and Pattern Recognition, Data Mining and Remote Sensing and Hyper-spectral Signal and Image Analysis.
Class Level Code Smells: Chernoff Face Visualization

Salman Abdul Moiz

School of Computer & Information Sciences, University of Hyderabad. Email: salman@uohyd.ac.in

Software systems that are poorly developed are prone to design anomalies. Such a system is said to be smelly system. The inherent market competition and pressure to develop a system within stringent deadlines results in an artifact which is not properly designed. This makes software difficult to maintain. The literature provides various mechanisms for detection of smells in the code. However it is noted that these smells are not known immediately as there is no proper mechanisms for its visualization. In this paper, an approach is presented to visualize the smelly classes in a better and effective way using chernoff faces. The proposed method uses fusion of Coefficient of variation and Information gain measure to select the best features for visualization. The results were simulated on two smelly classes namely Data class and God Class. The results shows that about 98% of the smells are visualized correctly.

Keywords: Code smell, refactoring, smelly classes, chernoff faces, God Class, Data Class.

Introduction

Maintenance is the inherent and one of the important phases of software development. However most of the time maintenance is either corrective, adaptive or perfective. The goal of the maintenance is to either address the bugs identified during testing, deploying systems on other environments or to add or modify the functionalities to the existing system. However a tested system may still contain anomalies as the elements of the code might be developed by violating the proven design principles. An element of a software smells when it violates the design principles and these anomalies hinder the maintenance activities. There are two phases to address the design principles violations. In the first phase the artifacts of the system i.e. package, class, methods etc. are checked to identify whether it has design issues which are termed as smells. In order to address these issues the code element has be modified. The process of modifying the software without affecting its behavior is known as refactoring. It improves the design of existing system (Fowler 1999). It is to be noted that the prerequisite for refactoring is that the source code is in proper working condition with test cases in place. The major phase is to detect the smells. Once detected it can be refactored.

Literature presents various mechanisms and tools to detect the smells and refactor them (Thanis 2017). Due to existence of diversified software’s and domains, most of the researches used their own approach in defining a smell. Further each tool adopts heuristic approaches to detect these smells. Even if it the smells are identified, they are not known immediately as there is no proper mechanism to visualize them. Visualization of smells helps in knowing the severity of the violation of the design principles.

Motivated by the smelly method visualization (Moiz 2020), in this paper an approach to visualize class level smells is proposed. The results are simulated on two smelly classes whose characteristics represents the facial features required for chernoff face visualization. Facial visualization represents the severity of smelly classes in the source code. Visualization the quality index of a code is one of the challenging tasks. As “face is the index of mind”, the visualization of smelly classes using facial representation will help in knowing the software design issues in the system. The advantage of such visualization is that the refactoring process can be started early and in turn it helps in maintenance phase. Once a regression testing is done, the system may go through the visualization process again to know the difference in the code quality that can be visualized using facial features. It is observed that 98% of the class level smells are visualized effectively using chernoff faces.

In this paper an approach is a mechanisms is proposed to identify the appropriate and relevant features (metrics) needed to visualize the class level smells using chernoff faces. The class and method level metrics are taken as the candidate features. A fusion of data driven, knowledge based and coefficient of variance is adopted to identify the candidate features needed for proper visualization of the smells. The metrics used in the literature are also visualized using chernoff faces which is required to validate the proposed approach.
The paper is structured into the following sections. Section-2 presents the state of art on smell detection techniques and visualization, section-3 presents an approach for smelly class visualization. It also presents the important metrics which are used in detection of God Class and Data Class smells. Section 4 presents the results and section 5 concludes the paper.

Related Work

Agile or Devops practices work on rapid development methods so that the design, build, test and release is carried out effectively within a given period of time. The pressure of market competition, timely delivery may result in the source code which doesn’t meet the design principles. Such anomalies are said to smell. Hence there is need to identify such smells. Smell detection techniques are divided into seven types (Kassentini 2014). This includes metrics based mechanisms, probabilistic techniques, search techniques, manual approaches, symptom techniques, cooperative approaches and visualization mechanisms.

Some of the visualization tools available in literature such as JDeodorant highlights the source lines of code which specifies the chunks responsible for a type of smelly code element. Further these visualization approaches requires human intervention to analyze the type of smell. They can’t specify the severity of the smell which can be better visualized using approaches like chernoff faces.

In metric based techniques for detection of smells, certain code level metrics are used as inputs to the rules that can detect whether a code smells. Search based techniques adopts machine learning mechanisms based on data available for the detection of code smells.

Fontana (2012) used metrics based mechanisms for detecting smells. This is achieved by using certain threshold limits for the metrics on the benchmark data set consisting of about several java projects. Classification algorithms were used for the detection of four code smells.

Chernoff faces were invented by Herman (1973) to visualize multivariate data using human faces. Ears, eyes, nose and mouth can represent values of given features. When the desired features are mapped to faces, it not only helps in proper visualization but also allows to identify even minor changes in the faces. The challenge is to select the most relevant and appropriate features that can be assigned to facial characteristics. Hai le (2000) specified about 10 basic facial patterns and also presented an approach to represent them as facial features.

Guggulothu (2019) used information gain measure in ranking the features for identification of four smells. However this approach uses the metrics which may not always be relevant. Sine only 15 features are needed for visualization using chernoff faces, Moiz (2020) ordered the features based on the attributes used in knowledge based systems used to detect smells (Fontana 2012) to visualize the code smells. However only method level smells where visualized.

In this paper a generic method for smelly classes visualization is presented and the same is applied on God Class and Data Class smells.

Visualization of god class and data class smells

Refactoring is the procedure of evolution of a software to address the smells identified in the system. The smelly classes present in the given system and can be identified using certain smell identification tools. However the knowledge based representation of these smells considers certain code level metrics as its inputs. This helps in identifying the candidate features required for detection of smelly classes.

God Class and Data Class

A God class is a large class which is capable of realizing many responsibilities. This class essentially has many methods. The metrics that helps in dealing with the size of methods are to be considered. In order to effectively realize the core functionality of a class, the getter and setter methods are to be excluded. If the software systems contains such classes, it is prone to design anomalies and such systems may fail in future. In order to address the god class smell, refactoring is to be performed on such classes to eliminate or reduce the design anomalies.

The refactoring process of a god class requires dividing the God class into multiple sub-classes with proper interfaces. However this refactoring shouldn’t violate the basic design principle which states that the coupling between the classes should be loose.

A data class is a class that contains only data i.e. it contains attributes. In addition it contains getter and setter methods which are used to access the data. They are usually used as data containers by other classes. The methods which requires access to such data may be available in other classes. This violates the basic design principle viz., encapsulation. There is a need to refactor such class to address the encapsulation design anomaly. This is achieved by Move Method and Extract Method. All the methods which accesses these data can be moved to the class holding the required data.

Smell Detection preliminaries

Literature presents certain smell detection tools which specifies the information about existence of a particular smell. However these tools doesn’t specify the reasons or conditions for occurrence of smells. Further few approaches (Arcelli 2015) (Moiz 2020) uses knowledge based systems for detection of smells which uses metrics. However these metrics alone may not be sufficient for identification of smells in heterogeneous systems. Guggulothu (2019) evaluated 64 features as metrics at various levels i.e package, class and method. The relevant features were selected as a function of the order of the information gain. Several object oriented metrics (Francesca 2015), (Arcelli 2015) (Moiz 2020) are proposed in literature which helps in identification of smells. Some of the metrics which are used in identification of smelly classes are as follows:

- LOCNAMM (Lines of Code without Accessor or Mutator Methods). It represents number of source lines of a class, excluding accessor or mutators but includes blank lines
and comments.

- **WMCNAMM (Weighted Methods Count of Not Accessor or Mutator Methods):** It is computed as the total sum of cyclomatic complexity of the methods of a class excluding accessor or mutator methods.
- **NOMNAMM (Number of Not Accessor or Mutator Methods):** This includes count of public and private methods of the class excluding accessor or mutator methods.
- **TCC (Tight Class Cohesion):** The tight class cohesion is the ratio of number of methods directly connected with other methods and the total number of connections between the methods.
- **ATFD (Access to Foreign Data):** This represents the number of attributes of other class that are either access directly or by invoking the accessor methods.
- **NOAM (Number of Accessor Methods):** The total number of get and set methods of a class.
- **WOC (Weight of a Class):** The total number of public methods divided by the number of public members.
- **NOAV (Number of Accessed Variables):** This represents the number of identifiers that are either accessed directly or by using the accessor methods.
- **FANOUT:** It represents number of called methods.
- **NOLV (Number of Local Variables):** This represents the total number of variables that are either accessed directly or by using accessor methods including global variables.
- **ATLD (Access to Local Data):** The number of attributes of current class accessed directly or by invoking accessor methods.
- **LOC (Lines of Code):** The number of lines of a method including blank lines and comments.
- **NOPA (Number of Public Attributes):** This specifies the total number of public attributes present in a class.

The code smell detection strategies are realized as knowledge systems (Fontana 2012) (Guggulothu 2019) and is represented using the following rules:

**God Class:**
\[
\text{LOCNAMM} \geq \text{HIGH}(176) \quad \text{WMCNAMM} \geq \text{MEAN}(22) \quad \\
\text{NOMNAMM} \geq \text{HIGH}(18) \quad \text{TCC} \leq \text{LOW}(0.33) \quad \text{ATFD} \geq \text{MEAN}(6)
\]

**Data Class:**
\[
\text{WMCNAMM} \leq \text{LOW}(14) \quad \text{WOC} \leq \text{LOW}(0.33) \quad \text{NOAM} \geq \text{MEAN}(4) \quad \text{WMCNAMM} \geq \text{MEAN}(22) \quad \text{NOPA} \geq \text{MEAN}(3)
\]

**Visualization of Smelly Classes**

The process of visualizing the class level smells is specified in the Fig.1.

---

**Fig.1: Smelly class visualization workflow**

In the first step, the class level smells which are to be visualized are defined. In this paper the God class and Data class smells are addressed. Several heterogeneous object oriented systems are then collected for identifying the smells. The features are then identified by finding the values of various metrics. These metrics are at different levels viz., package, class and method level. Smell detection tools are applied on same set of heterogeneous systems to identify whether the system is smelly or not w.r.t class level smells. The final labelling process is arrived after analyzing the results given by multiple tools, performing polling and if necessary a manual validation is done (Guggulothu 2019) (Umberto 2018). This helps in identifying the decision attribute. The data set now contains the conditional attributes evaluated as the code level metrics and the decision attribute specifying...
whether the corresponding code is smelly or not.

The information gain is computed for the features identified as metrics and are ordered to find the most relevant features. The relevance of these features are also compared with the knowledge based systems presented in the literature.

The coefficient of variation (CV) of these features is also computed and is arranged in the descending order. Since 15 features are required for visualization using chernoff faces, initially the outliers are identified. The CV values of the corresponding top 15 features identified using information gain defines the boundary of the subset of features. The features which are outside the specified range are deleted.

Later the features are further refined by removing package level and attribute level metrics as the aim is to visualize smelly classes. The resultant set of features are further refined if needed based on the domain knowledge experts feedback. The final resultant metrics are mapped to the facial characteristics and are then visualized.

Results & Analysis

In this paper, two datasets of god class and data class respectively are considered from Fontana (2012). Out of total 420 instances 2/3 of them are represents negative decision attributes that is non-smelly and remaining represents positive instances i.e the smelly classes.

The information of gain of these features are computed and are ordered to find the relevant features (Guggulothu 2019). Similarly, the coefficient of variance (CV) of these features is computed and ordered. Though 18 features can be used for visualization in chernoff faces. However only 15 features are used in this paper as the faces() function in R studio takes maximum of 15 features.

In case of God class, the CV value of the top 15 features selected by information gain is computed which is between 329.6 and 155. All the features whose CV is out of this range are removed which are outliers. Then the package level features and attribute level features are discarded. The remaining 15 features are mapped to the attributes required for chernoff face visualization.

Fig. 2, depicts the visualization of non-smelly god classes using the proposed method of feature selection.

The common features which can be visualized from the Fig. 2 includes green hat domination in non-smelly classes. The face visualization helps in easily identifying the design anomalies in the given software system which is assessed for god classes. Fig. 3 specifies few instances of smelly god classes using the proposed method.

It is observed that the smelly god classes are either dominated by red color faces or includes big faces with wide caps or horns. This type of visualization helps in identifying the smelly god classes.

The rule based systems used to identify the god class (Fontana 2012) uses five features namely ATFD_type, NOMNAMM_type, LOCNAMM_type, TCC_type and WMCNAMM_type. These attributes are repeatedly mapped to the required 15 characteristics of faces. Figure 4 presents the visualization of few of the non-smelly god classes using the rules present in the literature. Similarly figure 5 depicts the visualization of subset of smelly god classes using the rules.

The proposed method is validated using visualization of smells based on the features used in smell detection rules. However the visualization using the existing rules considers only 4 to 5 features of faces whereas the proposed method of visualization uses 15 different features based on the proposed method. For example in case of God class, the code metrics features are mapped to the facial features using existing rules as follows:

- “height of face “ “ATFD_type”
- “width of face “ “NOMNAMM_type”
- “structure of face” “WMCNAMM_type”
- “height of mouth “ “LOCNAMM_type”
- “width of mouth “ “TCC_type”
- “smiling “ “ATFD_type”
- “height of eyes “ “NOMNAMM_type”
- “width of eyes “ “WMCNAMM_type”
"height of hair  " "LOCNAMM_type"
"width of hair   " "TCC_type"
"style of hair   " "ATFD_type"
"height of nose  " "NOMNAMM_type"
"width of nose  " "WMCNAMM_type"
"width of ear   " "LOCNAMM_type"
"height of ear   " "TCC_type"

Similarly experimentation is performed for visualization of data class smells.

In the data set available for detection of Data class, the CV value of the top 15 features selected by information gain is between 245.145 and 114.414. All the features whose CV is not in the identified range are treated as outliers and are removed. Then the package level features and other features not relevant to a data class smell are discarded. The remaining 15 features represents the characteristics required for chernoff faces for the purpose of visualization.

Figure 6, represents the visualization of non-smelly data classes using the proposed method of feature selection.

![Fig. 6: Non-smelly Data classes](image)

It is observed that the green hat dominates the non-smelly classes. Even a wide hat is also noted, but it is usually observed that in non-smelly classes it is white in color. Figure 7 gives the visualization of few of smelly data classes using the proposed approach.

![Fig. 7: Smelly Data classes](image)

The visualization of smelly data classes are either dominated by red color faces or includes big faces with wide caps or horns and these caps or horns are not in white color. The type of visualization helps in easily identifying the anomalies in the system which can be addressed during the maintenance phase.

The rule based systems used to detect data class smells (Umberto 2018) (Arcelli 2015) uses four features namely WMCNAMM_type, WOC_type, NOAM_type and NOPA_type. These features are repeatedly mapped to the required 15 characteristics of faces. Figure 8 and 9 respectively represents the visualization of subset of non-smelly and smelly data classes using the rules presented in the literature.

![Fig. 8. Non-smelly Data classes using rules](image)

![Fig. 9. Smelly Data classes using rules](image)

The visualization of rule based smell detection helps in validation of the proposed method of visualization. However the proposed method has more visualization features as it considers all the 15 features of the faces for visualization of the smell.

The results were simulated for 420 instances which represents a class in a system. For each such features initially 64 features were considered. These features are code metrics which are at various level of granularity i.e. project, package, class and method levels respectively.

**Conclusion**

Smell detection is the preliminary for refactoring the source code to make the maintenance process simple. The rapid application approaches lacks the instant feedback of the design anomalies in the system as there is no proper mechanisms for visualization of such smells.

In this paper an approach is presented for the visualization of the smells. It uses the information gain and Coefficient of variation for better visualization of smelly classes. The proposed approach was simulated on data glass and god class and it is observed that the facial features assists in understanding the severity of the smells.

The proposed approach is validated against the visualization of features extracted from rules proposed in literature for identification of smells. In the proposed approach metrics of software systems were ordered based on the coefficient of variation of features used in the rules.

Though the total number of features at the Project, package, class and method level are 64, the rules specified in literature uses 4 to 5 such features. However the proposed method of smelly class visualization uses all 15 feature required for visualization using chernoff faces. Hence the proposed method is capable to visualize the severity of the smells better as compared to the rules proposed in literature.

In future the package level and project level smells can be visualized and a holistic view of smells of packages,
class and method level can be presented. Further the order of refactoring’s can be suggested so that if certain smell is addressed first then some of the dependent smells may be addressed implicitly.

References

13] Bremen, Germany: IEEE, Oct. 2015, pp. 16–24, in conjunction with ICSME.

About the Author

Dr. Salman Abdul Moiz is working as an Associate Professor in School of Computer & Information Sciences at University of Hyderabad. Earlier he worked as Professor in GITAM University & MVSR Engineering College. He also worked as Research Scientist at Centre for Development of Advanced Computing Bangalore. His areas of interests include Software Engineering, E-Learning technologies and mobile database systems.
Paragraph level semantic plagiarism detection for Marathi language

Ramesh Ram Naik, Maheshkumar B. Landge, C. Namrata Mahender
Department of CS and IT, Dr. B. A. M. University, Aurangabad-431004 (MS), INDIA

Plagiarism is stealing of information from online sources and presenting that information as one’s own without giving proper acknowledgement to original authors. There are different types of plagiarism like copy and paste, idea, metaphor, self, translation, mosaic, structural plagiarism. There are two types of plagiarism detection techniques namely: i) Extrinsic plagiarism detection techniques in which plagiarism identified on basis of reference corpus and ii) Intrinsic plagiarism detection techniques in which plagiarism identified on the basis of writing style of author. As compare to other Indian languages, a very few work has been done on plagiarism detection for Marathi language. This paper is mainly focused on semantic plagiarism detection for Marathi language at paragraph level. Marathi text corpus and Marathi wordnet has been developed. Marathi wordnet is used to find the semantic similarity percentage.

Keywords: Semantic plagiarism detection, Marathi wordnet, Marathi language.

1. Introduction

Plagiarism is the use of ideas, concepts, words, or structures without appropriately acknowledging the source to benefit in a setting where originality is expected [1].

There are two techniques for plagiarism 1) Extrinsic plagiarism detection and 2) Intrinsic plagiarism detection. In extrinsic plagiarism detection reference corpus is used to detect plagiarism and intrinsic plagiarism detection writing style is used to identify the plagiarism. In this paper our main work is focused on semantic plagiarism detection for Marathi language. If any Author changes the meaning of text using synonyms, without changing the position of words. This is known as semantic plagiarism. Semantic processing determines the possible meanings of a sentence by focusing on the interactions among word-level meanings in the sentence [2].

2. Literature survey


Urvashi Garg et. al. [9] developed software for Hindi plagiarism detection tool named as Maulik. This tool is used to detect the plagiarism in Hindi document. They used N-gram technique. More work has been done for plagiarism detection in foreign languages like English, French, Dutch, Greek, Persian etc. In India, few researchers are working with plagiarism detection in regional languages like Hindi, Malayalam, Urdu, Bengali, Punjabi. For Marathi language few researchers has started working for plagiarism detection. Shenoy [10] used Marathi language dataset, in which they used some manual and artificial paraphrases for testing. The fuzzy semantic similarity search model and the Naïve Bayes model are used.

3. Paragraph level semantic plagiarism detection model [PLSP]

The Paragraph level semantic plagiarism detection model [PLSP] is designed for Marathi text in the presented work for this paper. The procedure of PLSP is discussed in detail below, but the actual work does not start from PLSP model instead it starts from corpus development as even today not much work is seen on Marathi text plagiarism. A sample corpus is developed with 10 paragraphs. The system is also linked with wordnet of Marathi [11]. Then 50 candidates of native speakers in the age group of 20-27 at random were
asked to write these paragraphs in their own words. Actually the candidates were requested to create a summary out of the original paragraph in a sense as they need to reuse it in some other work. Figure1 shows original paragraph and figure2 shows sample paragraph written by a candidate.

Then preprocessing is done on original as well as collected summaries from the candidates. In preprocessing, the text are tokenized, punctuations are removed i.e. in present work ''' ! ( )- [ ]{};:'".,<>./?@#$%^&*_~''' are removed and stopwords are collected. Stop words are basically a set of commonly used words in any language. Figure4 shows sample after tokenization of original paragraph, figure6 shows sample after punctuation removal and table1 shows sample of stopwords.
A) **PLSP model procedure**: The input to the model are the files which are cleared from punctuation and stopwords. Figure 7 shows sample of input files to PLSP model.

**Step 1:**

The input file is broked into sentences and each sentence is subdivided into N-gram for further processing. In present work unigram, bigram, and trigram of the sentences are generated. Unigram is each individual work present in the sentence, bigram are the pair of words in the sentence and trigram are group of three words. Trigram actually gives the probability of word with its predecessor and successor word. We tried with fourgram and five gram too, but it didn’t show much difference thus upto trigram the combinations are only considered. Table-2 show sample of unigram, bigram, and trigram

**Step 2:**

The original paragraphs unigram, bigram and trigram data along with the candidates unigram, bigram and trigram for the same paragraph is matched with wordnet. As in wordnet for each word there is a synonym set called synset, representing the meaning (relation). WordNets convey different information about the words like which word has multiple meanings, what are the relationships between the words and synonym sets etc. The lists of sample words from wordnet are showing in Table 3.
Step 3:
The matched results of candidate data are compared with the original paragraph and a similarity matrix is calculated. The resultant similarity matrix representing the percentage of similarity between candidate file and original file and semantic similarity percentage of 500 summary files of 50 candidates is shown in Table 4.

<table>
<thead>
<tr>
<th></th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
<th>P9</th>
<th>P10</th>
</tr>
</thead>
<tbody>
<tr>
<td>CD1</td>
<td>2.4</td>
<td>1.83</td>
<td>4.03</td>
<td>0.84</td>
<td>2.76</td>
<td>2.04</td>
<td>1.21</td>
<td>3.07</td>
<td>4.18</td>
<td>0.7</td>
</tr>
<tr>
<td>CD2</td>
<td>1.64</td>
<td>3.61</td>
<td>1.52</td>
<td>2.91</td>
<td>2.84</td>
<td>2.06</td>
<td>2.59</td>
<td>4.23</td>
<td>11.82</td>
<td>4.95</td>
</tr>
<tr>
<td>CD3</td>
<td>3.6</td>
<td>0</td>
<td>0</td>
<td>3.2</td>
<td>2.14</td>
<td>1.3</td>
<td>1.91</td>
<td>1.44</td>
<td>2.29</td>
<td>5.53</td>
</tr>
<tr>
<td>CD4</td>
<td>4.34</td>
<td>4.23</td>
<td>3.1</td>
<td>0.89</td>
<td>1.27</td>
<td>2.29</td>
<td>1.62</td>
<td>10.81</td>
<td>2.46</td>
<td>3.67</td>
</tr>
<tr>
<td>CD5</td>
<td>2.91</td>
<td>2.74</td>
<td>1.33</td>
<td>2.17</td>
<td>1.34</td>
<td>2.23</td>
<td>2.44</td>
<td>1.29</td>
<td>2.18</td>
<td>3.45</td>
</tr>
<tr>
<td>CD6</td>
<td>2.16</td>
<td>2.03</td>
<td>1.56</td>
<td>2.95</td>
<td>1.43</td>
<td>3.29</td>
<td>3.29</td>
<td>1.42</td>
<td>1.88</td>
<td>2.88</td>
</tr>
<tr>
<td>CD7</td>
<td>1.36</td>
<td>3.46</td>
<td>0.9</td>
<td>6.14</td>
<td>0.61</td>
<td>3.48</td>
<td>4</td>
<td>12.28</td>
<td>17.82</td>
<td>3.73</td>
</tr>
<tr>
<td>CD8</td>
<td>4.06</td>
<td>3.31</td>
<td>1.69</td>
<td>3.5</td>
<td>2.58</td>
<td>2.55</td>
<td>1.7</td>
<td>2.06</td>
<td>4.2</td>
<td>3.03</td>
</tr>
<tr>
<td>CD9</td>
<td>4.38</td>
<td>2.58</td>
<td>2.16</td>
<td>1.51</td>
<td>3.65</td>
<td>2.47</td>
<td>2.96</td>
<td>1.3</td>
<td>1.78</td>
<td>3.09</td>
</tr>
<tr>
<td>CD10</td>
<td>1.78</td>
<td>1.32</td>
<td>1.79</td>
<td>3.05</td>
<td>1.33</td>
<td>2.37</td>
<td>3.24</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CD11</td>
<td>3.03</td>
<td>2.95</td>
<td>2.09</td>
<td>2.35</td>
<td>2.65</td>
<td>2</td>
<td>2.78</td>
<td>0.47</td>
<td>1.88</td>
<td>1.64</td>
</tr>
<tr>
<td>CD12</td>
<td>2.87</td>
<td>1.1</td>
<td>1.56</td>
<td>2.66</td>
<td>2.56</td>
<td>1.76</td>
<td>4.19</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
In above table 4 from CD1-CD50 represents candidate1 to candidate 50 and P1-P10 represents paragraphs from 1 to 10.

<table>
<thead>
<tr>
<th>Paragraphs</th>
<th>Average semantic similarity percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>3.6668</td>
</tr>
<tr>
<td>P2</td>
<td>3.0462</td>
</tr>
<tr>
<td>P3</td>
<td>4.682</td>
</tr>
<tr>
<td>P4</td>
<td>3.3432</td>
</tr>
<tr>
<td>P5</td>
<td>3.5644</td>
</tr>
<tr>
<td>P6</td>
<td>3.06642</td>
</tr>
<tr>
<td>P7</td>
<td>5.0698</td>
</tr>
<tr>
<td>P8</td>
<td>5.7254</td>
</tr>
<tr>
<td>P9</td>
<td>6.6724</td>
</tr>
<tr>
<td>P10</td>
<td>4.8234</td>
</tr>
</tbody>
</table>

4. Result:

Original Paragraphs files from database are having maximum length of 21 sentences and minimum length of 3 sentences. Sentences are having maximum length of 10 words and minimum sentence length is of 4 words. From 500 summary files written by candidates maximum length of paragraph is 24 sentences and minimum length is of 6 sentences. Sentences are having maximum length of 15 words and minimum sentence length is of 7 words.

Following Table 5 shows Average semantic similarity percentage:

Table 5: Average semantic similarity percentage

<table>
<thead>
<tr>
<th>Paragraphs</th>
<th>Average semantic similarity percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>3.6668</td>
</tr>
<tr>
<td>P2</td>
<td>3.0462</td>
</tr>
<tr>
<td>P3</td>
<td>4.682</td>
</tr>
<tr>
<td>P4</td>
<td>3.3432</td>
</tr>
<tr>
<td>P5</td>
<td>3.5644</td>
</tr>
<tr>
<td>P6</td>
<td>3.06642</td>
</tr>
<tr>
<td>P7</td>
<td>5.0698</td>
</tr>
<tr>
<td>P8</td>
<td>5.7254</td>
</tr>
<tr>
<td>P9</td>
<td>6.6724</td>
</tr>
<tr>
<td>P10</td>
<td>4.8234</td>
</tr>
</tbody>
</table>

According to the graph shown in Figure 7, semantic plagiarism has been detected in a 500 Summary files from database with different percentage, where the highest average semantic similarity percentage is 6.67% and lowest percentage is 3.0%.

5. Conclusion

Plagiarism is using someone’s work as it is without giving proper acknowledgement. Plagiarism is a serious issue in academic area. There are two techniques used for plagiarism detection i.e. extrinsic and intrinsic. This paper is focused on detecting semantic plagiarism using wordnet. Wordnet is used to detect semantic plagiarism by using synonyms of words in text file. The graph of semantic similarity percentage shows the highest average semantic similarity percentage is 6.67% and lowest percentage is 3.0% from 500 sample files in database.

Acknowledgement

We would like to acknowledge and thanks to CSRI DST Major Project sanctioned No.SR/CSRI/71/2015 (G), Computational and Psycholinguistic Research Lab Facility supporting to this work and Department of Computer Science and Information Technology, Dr. Babasaheb Ambedkar Marathwada University, Aurangabad, Maharashtra, India.

References

1] Fishman, T. “We Know It When We See It” Is Not Good Enough: Toward a Standard Definition of Plagiarism That Transcends Theft, Fraud, and Copyright. 2009.
About the Author

Ramesh R. Naik has completed M.Sc.(Computer Science), M.Phil. (Computer Science), currently pursuing the Ph.D. degree in Computer Science in Department of Computer Science and IT, Dr. Babasaheb Ambedkar Marathwada University, Aurangabad, Maharashtra, India. He is presently working as Assistant Professor in Department of Computer Science at Deogiri College, Aurangabad. His research interest includes Natural Language Processing, Artificial Intelligence, Machine Learning.

Maheshkumar B. Landge has completed M.Sc.(Computer Science), M.Phil. (Computer Science), currently pursuing the Ph.D. degree in Computer Science in Department of Computer Science and IT, Dr. Babasaheb Ambedkar Marathwada University, Aurangabad, Maharashtra, India. He is presently working as DST Project Fellow under the project entitled “To Explore the Impact of SNS on Behavioral, Physical and Mental Development of Indian Preadolescents and Adolescents” sanctioned by DST under the guidance of Dr. C. Namrata Mahender, Assistant Professor, Dept. of CS and IT, Aurangabad. His research of interest includes Natural Language Processing, Computational Linguistics, Machine Learning.

Dr. C. Namrata Mahender has qualified NET exam in 2006 and completed Ph.D. from Dr. B. A. M. University Aurangabad in 2007. She is working as a Assistant Professor in Department of Computer Science and IT, Dr. Babasaheb Ambedkar Marathwada University, Aurangabad(MS) India. She is a life member of ISCA, IETE, IAEng, and CSTA. She has published more than 85 research papers in National and International reputed journals/conferences. Her research of interest includes Natural Language Processing, Artificial Intelligence, Psychoanalysis and Computational Intelligence, Computational Linguistics, Pattern Recognition, Image Processing, Artificial Neural Network, Fuzzy Logic.

Call for Paper for CSI Journal of Computing

Original Research Papers are invited for the CSI Journal of Computing, published online quarterly (e-ISSN: 2277-7091) by the Computer Society of India (CSI). The Journal of Computing, offers good visibility of online research content on computer science theory, Languages & Systems, Databases, Internet Computing, Software Engineering and Applications. The journal also covers all aspects of Computational intelligence, Communications and Analytics in computer science and engineering. Journal of Computing intended for publication of truly original papers of interest to a wide audience in Computer Science, Information Technology and boundary areas between these and other fields. The articles must be written using APA style in two columns format. The article should be typed, double-spaced on standard-sized (8.5” x 11”) with 1” margins on all sides using 12 pt. Times New Roman font and 8-12 pages in length. The standard international policy regarding similarity with existing articles will be followed prior to publication of articles. The paper is to be sent to Prof. A K Nayak, Publisher, in the email id: csi.journal@csi-india.org, aknayak@iibm.in with a copy to Professor Ratnadeep R. Deshmukh, in the email id: rrdeshmukh.csit@bamu.ac.in, Editor in Chief, CSI Journal of Computing.

Prof. A K Nayak
Publisher
Guidelines to Authors
General Submission Guidelines
Journal of Computing
Published by the
Computer Society of India

General
- The manuscript submitted should be author’s original work; citations to others work should have appropriate citations with due credit and must adhere to standard copyright rules. It is the responsibility of the authors to obtain the necessary disclosure permissions for the material used (be it Government, Corporations, Societies, Publishers or authors) and not the CSI (CSI has a zero tolerance policy on plagiarism; cf. http://www.csi-india.org/policies) The paper should not have been published earlier and must not be under consideration for publication elsewhere.
- Regular papers will be generally 15 to 20 pages in the double column of size 8.5” x 11” (or 21.6cm x 27.9cm) each column of 3.5” (width 6.45cm).

Manuscript Formats
a. Full papers generally consist of the title, author affiliation, footnote (including any financial support acknowledgment), abstract, introduction, body, conclusions, reference list, list of figures and table captions, and original figures and tables for reproduction.
b. A paper may also include appendixes, a glossary of symbols, and an acknowledgment of nonfinancial support. All manuscript pages, footnotes, equations, and references should be labelled in consecutive numerical order. Illustrations and tables should be cited in text in numerical order.
c. Abstract: The abstract should be limited to 50–200 words and should concisely state what was done, how it was done, principal results, and their significance. The abstract will appear later in various abstracts journals and should contain the most critical information of the paper.
d. References: A numbered list of references must be provided at the end of the paper. The list should be arranged in alphabetical order. List only one reference per reference number. In text, each reference number should be enclosed by square brackets. Footnotes or other words and phrases that are not part of the reference format do not belong on the reference list.
e. Section Headings: Primary section headings within papers are enumerated by Roman numerals and are centered above the text. For the purpose of typing the manuscript only, primary headings should be capital letters.

Peer Review
- After the Editor/Editor-in-Chief of a publication determines that a paper is suitable for his/her publication, it will be forwarded to a group of reviewers (generally at least three) selected for their expertise in a given field. During this process, an author is often asked to expand, rewrite, or explain further the content of his/her paper. It is not uncommon that an author is asked to provide another draft with the suggested changes for further review.
- Generally it may take up to six months from the time of submission.
- Authors have the right to withdraw the manuscript at any stage of the review process.

Final Acceptance:
- Once a manuscript has received the final approval of the reviewers and Editor-in-Chief, the author will be notified and sent an CSI Copyright Form. He/she will be asked to prepare the manuscript for final electronic publication along with a reprint/colour reproduction invoice.
- Final Manuscript Submission: For accurate and efficient transferal of your manuscript, particularly those containing extensive mathematics, it is recommended that the authors may use TEX or LATEX programs. A CSI LATEX style file can be found at www.csijournal.org/guidelines.html. The .cls file can be downloaded here. Also, in your file use CSI header “CSI Journal of Computing” using markboth as indicated in the illustrating file. You may use the bibliography style as in IEEE (number the references) style format (TRANS-JOUR). Doc users can use the style guide provided at this address. For headers, use the CSI Journal of Computing. The copyright form can be downloaded here. Each author needs to submit a passport photograph in jpg format, with photo size less than 30 KB. Also, a short bio (about 6 lines) need to be provided for each author.

Page Charges
- No page charges.
- Colored reproduction of figures, photographs at no cost for the web version. For print version, actual print charges to be borne by the authors. If the cost is not borne, the same would be reproduced in black and white. At the time of proofs being sent, appropriate invoice will be sent.
The best paper in each volume of the CSI Journal of Computing will be given an award in one of the CSI sponsored conferences.

The details will follow.