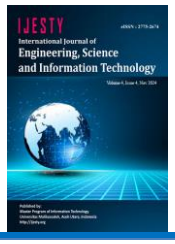
International Journal of Engineering, Science and InformationTechnology
Volume 4, No. 4 (2024) pp. 272-279
ISSN 2775-2674 (online)
Website: http://ijesty.org/index.php/ijesty
DOI: https://doi.org/10.52088/ijesty.v4i4.677
Research Paper, Short Communication, Review, Technical Paper



Building a Web Crawler for Text Data Indexing on Online Newspaper Web

Jamaludin Hakim[1]*, Andrian Sah[1], Siti Nurhayati[1], Wahyu Ciptaningrum[2], Damar Suryo Sasono[2]

[1]Department of Information System, Universitas Yapis Papua, Indonesia
[2]Department of Informatics, Universitas Yapis Papua, Indonesia

**Corresponding author Email: j2hakim@gmail.com*

The manuscript was received on 20 April 2024, revised on 28 August 2024, and accepted on 18 November 2024, date of publication 10 December 2024

Abstract

The Internet has become a vast repository of information, often filled with distractions that can hinder the user experience. News content, for example, is usually interspersed with advertisements that interrupt the flow of reading. In addition, the fast pace of news publication is also a challenge, with potentially more than 50 new articles appearing in 20 minutes. This high-speed data flow is valuable for various applications, including Social Media Analytics Services. In this context, the speed and efficiency of data acquisition (crawling) and processing (scraping) are critical. These processes must be optimized to ensure comprehensive data collection without gaps, focusing on the latest information. To meet this need, we propose developing an application capable of capturing news data in its entirety, minimizing the risk of missing important information. At the core of this solution is a web crawler- a sophisticated program designed to automatically browse the hyperlink structure of the web, systematically downloading linked pages to local storage. This crawling methodology is often the basis for web mining initiatives and search engine development. Since web information is distributed across billions of pages hosted on millions of servers worldwide, our application utilizes the PHP programming language to capture and process this data effectively. The main goal is to present pure news content to users without any irrelevant elements. We use a Data Flow Diagram (DFD) to model the system architecture and data flow. This approach provides a clear visualization of how web users can navigate through hyperlinks to efficiently access the desired news information. By implementing this system, we aim to improve the user experience of consuming news content, facilitate more effective data analysis, and contribute to the broader web information search and processing field.

Keywords: *Web Crawler, Scraping, News Content, Data Flow Diagram, PHP.*

1. Introduction

The rapid development of the Internet has resulted in an exponential growth in the number of websites, creating considerable challenges in searching and indexing relevant information, especially for online news [1]. With unique social and political dynamics, the need for fast and efficient access to local news is even more crucial [2]. Web crawlers, as automated programs that explore the hyperlink structure of the web, are a vital solution to this challenge. Crawlers collect data and enable efficient indexing of text content, making it easier for users to find the information they need [3].

This research aims to build an optimized web crawler for indexing text data on online news sites in a City. Considering the unique characteristics of local news content and website structure in the City, the crawler will be designed to achieve maximum data collection and indexing efficiency. The development of this web crawler will not only facilitate better access to online news in the City but also open up opportunities for further studies on information dissemination patterns, media content analysis, and a better understanding of the digital landscape in the region. The results are expected to improve information accessibility, support more in-depth news analysis, and ultimately contribute to a better information ecosystem in the City.

2. Literature Review

2.1. Web Crawler

A web crawler is a system that explores the hyperlink structure of the web from a start or seed address and regularly visits web addresses within a webpage [4]. Search Engines are one example of an extensive system that uses crawlers to traverse the Internet continuously to find and retrieve as many web pages as possible [5]. The following is the process that web crawlers perform [6]:

1. Downloading webpage
2. Parses the downloaded webpage and retrieves all links.
3. For each retrieved link, repeat the process.

In the crawling process, the term spider is used. Spiders are tasked with collecting information about blog sites or websites. Spiders collect information from links, HTML structures, meta tags, titles, and text content. Spiders can crawl blog sites or websites that use/have a robots.txt file. Robots.txt contains a script that the spider will then translate as a command to collect the information mentioned. And robots.txt will also make it easier for spiders to collect data. The crawling process is critical. The search engine will not recognize the blog site or website if the crawling process does not run smoothly [7].

The Web Crawler will be given an initial URL to search and perform the parsing stage. The parsing stage is carried out to retrieve links, from which the links will be crawled again, and to retrieve news content, which is then carried out in the preprocessing stage. The preprocessing stage will carry several stages: HTML tag removal, tokenizing, filtering, and stemming [8]. The following is a picture of the web crawler architecture:

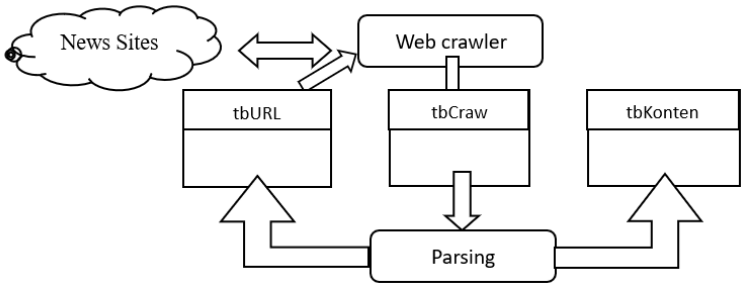


Fig 1. Web Crawler Architecture

Figure 1 shows the flow of the crawling process, where the link is the seed or root. Before crawling, the depth or max level will be determined first, which helps limit where the crawling process will occur.

2.2. Data Mining

Data mining is software that discovers hidden patterns or relationships in large databases and generates rules to predict future behavior [9]. Data mining is often called Knowledge Discovery in Database (KDD). KDD is an activity that includes collecting and using data historically to find regularities, patterns, or relationships in large data sets [10]. The stages of the data mining process are as follows:

1. Selection: This stage aims to get text data relevant to the analysis task in the next stage. At this selection stage, the selection and filtering of raw data will be carried out into target data through several selection stages, namely sampling, denoising, and feature extraction.
2. Preprocessing includes removing duplicate data, checking for inconsistent data, and correcting errors in the data, such as printing errors. An enrichment process also enriches existing data with other relevant data or information needed for KDD, such as external data.
3. Transformation coding is transforming the data that has been selected so that the data is suitable for the data mining process. The coding process in KDD is creative and depends on the type or pattern of information to be searched in the database.
4. Data mining is finding interesting patterns or information in selected data using specific techniques or methods. Techniques, strategies, or algorithms in data mining vary widely. The selection of an appropriate method or algorithm depends mainly on the objectives and the overall KDD process.
5. Interpretation / Evaluation: from the results of the data mining process above, an evaluation process will be carried out to obtain attractive, more understandable, and valuable knowledge for interested parties. This stage includes checking whether the patterns or information found conflict with previously existing facts or hypotheses.
6. Knowledge presentation presents extracted knowledge to users using visualization and representation techniques.

2.3. Dynamic Programming with PHP

PHP (Personal Home Page) is a programming language to build a dynamic website [11]. PHP is integrated with HTML code. HTML is used as a builder or foundation of a web layout framework, while PHP functions as a process so that a web will be easy to maintain with PHP [12]. PHP runs on the server side, so PHP is also known as the Server Side Scripting language, meaning that in every / to run PHP, you need a web server to run it [13].

PHP is a programming language used to create website programs where the program code that has been made is compiled and run on the server side to produce dynamic web pages. PHP was created in 1994 by Rasmus Lerdorf [14]. PHP was initially established for a Personal Home Page. PHP is called a hypertext processor because it has many benefits and can be developed well. PHP is an open-source software. Writing PHP program code is integrated with HTML, which runs on the server side. All syntax written is fully executed on the server; only the results are sent to the browser [15]. PHP is a programming language suitable for creating dynamic web applications such as CMS because it has high performance, is easy to learn, is multi-platform, is secure, is open source, and is easy to connect with various database systems [16].

PHP is interpreter programming, which translates lines of source code into machine code that the computer understands directly when the lines of code are executed. PHP is called server-side programming because the entire process runs on the server. PHP is a language with open copyright, also known as Open Source, which means that users can develop PHP function codes according to their needs. PHP programming can be written in two forms: writing lines of PHP code in single files and writing PHP code on HTML pages (Sibero, 2011). PHP (Hypertext Preprocessor) is a programming language that can only run on the server side (Server Side Scripting). This means that processes made with PHP will not run without a web server. PHP is used to build web-based applications so that the web can be used dynamically, such as adding, changing, reading, and deleting content. PHP is integrated with HTML code. PHP does not replace the central role of HTML as the foundation of the web framework but completes the void. HTML is a programming language used to build the framework or foundation of the web. Meanwhile, PHP is a programming language used to process the actions contained in web content.

2.4. Context Diagram

A context diagram is the top-level Data Flow Diagram (DFD) of an information system that depicts the system in a loop that represents the entire process in a system [17]. Depicting a context diagram consists of the following:

1. Draw the system as a circle and name the system.
2. Draw a box of external entities and name the entity.
3. Create a data flow from each external entity.

DFD (Data Flow Diagram) is a tool to describe or create a model that allows system professionals to tell the system as a network of functional processes connected by data flow, both manually and computerized [18]. In general, this data flow diagram is a network that describes an automated/computerized system, manualization, or a combination of the two, whose depiction is arranged in a collection of system components interconnected according to the game's rules. The advantage of DFD users is that it is possible to describe the system from the highest level and then reduce it to a lower level (decomposition). Meanwhile, the disadvantages of using DFD do not show looping, decision processes, and calculation processes [19].

DFDs provide additional information used during information domain analysis and serve as the basis for function modeling. DFDs can be used to present a system or software at any level of abstraction. DFDs can be partitioned into levels representing incremental information flows and idealized functions. DFDs provide a mechanism for both functional modeling and information flow modeling. DFDs are also one of the most frequently used modeling tools, especially when system functions are more critical and complex than the data manipulated by the system. In other words, DFD is a modeling tool that emphasizes only the system's functions. DFD is also a data flow-oriented system design tool with the concept of decomposition, usually used for depicting system analysis and design that system professionals easily communicate to users and programmers.

The steps in creating a data flow diagram are divided into 3 (three) stages or levels of DFD construction, which are as follows [20]:

1. Context diagram: This diagram describes the source and destination of the data to be processed; in other words, the diagram illustrates the system in general / globally from the existing system as a whole.
2. Zero diagram: This diagram describes the process stages in a more detailed context diagram.
3. Detail diagram: this diagram describes the data flow in more detail from the process stages in the null diagram.

3. Methods

This analysis method stage is the stage of analyzing the system to be built. After the analysis is obtained, the next step is to make an analysis result. The analysis results will be a reference for the design of the built system. The needs needed to build the desired system are divided into two parts, namely input needs and output needs, as follows:

1. Input requirement
   In this case, this is the input of the web address of the Online Newspaper in City.
2. Output requirement
   This case consists of a crawling process page display and a text file display.

At this stage, a system design will be carried out to carry out the crawling process on the Online Newspaper News site in City in the form of a link or URL. The design will be carried out using the PHP programming language. The design results using the Brute Force method are implemented using the PHP programming language. To determine the level of success of the program created.

A flowchart is a technique that makes it easier for us to program, in this case, making it more accessible in the sense of anticipating that no program components are left behind. Below is an overview of the crawler system flowchart that will run as follows:

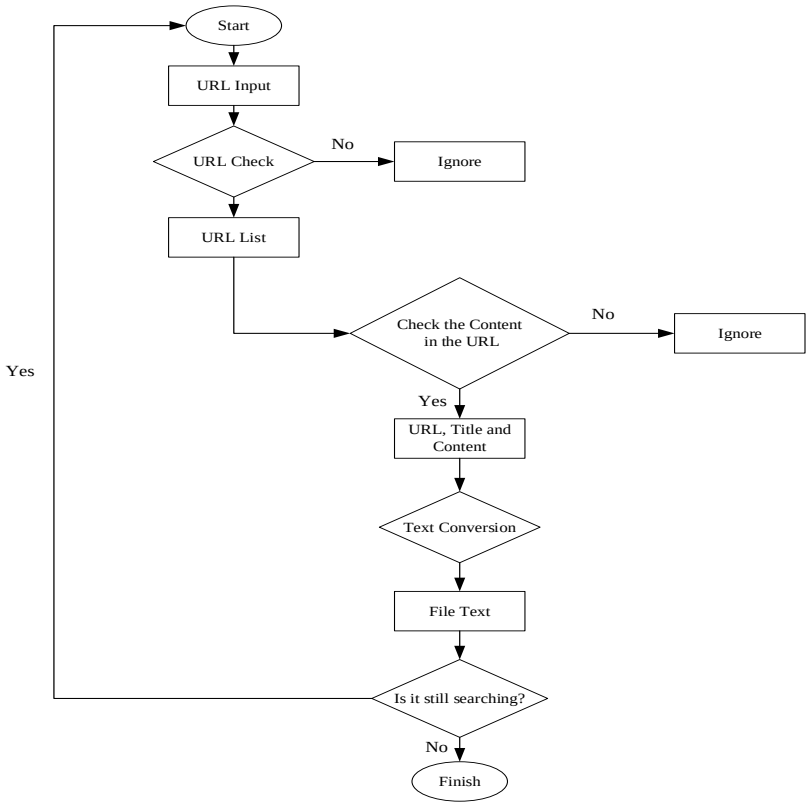


Fig 2. Crawler Flow Chart

The system starts with the user entering a website address or URL to search. The system then checks for URLs within the site; if found, the system displays a list of existing URLs. Next, the system performs further checks to find the news content in each detected URL. If news content is found, the system displays the URL, title, and content. The user can convert or download the displayed information in .txt file format at this stage. Once this process is complete, the user can perform a new search by returning to the URL input stage or end the process if no further search is desired. This workflow allows users to efficiently browse, extract, and store news content from targeted websites.

4. Result and Discussions

In this system analysis, the decomposition of an intact information system into its parts will be carried out to identify and evaluate problems so that weaknesses are found. The obstacles that occur and the expected needs so that improvements can be proposed.
There are two types of needs based on the needs that will be applied to this system. These two needs have different access rights, namely the needs of a user and an admin.

1. User Needs
   a. Search for the website address, which functions to search for the website address the user wants to crawl.
   b. View the link index, which functions to view the list of links on the website.
2. Admin Needs
   a. Input replacement words are functions that replace words that have been crawled on a website.
   b. Editing root words and functions to change root words crawled on a website.

Several steps will be taken when designing this web crawler. The following are the system design steps:

1. Describe the context diagram of the book search system.
2. Describe the DFD (Data Flow Diagram) to explain more details of the context diagram and to know the data flow in the system.
3. Describe the ERD (Entity Relationship Diagram) to explain the relationships between entities in the system.

The context diagram broadly explains the input, process, and output generated from the system being built. In this system, two entities interact with the system: the user and the admin.

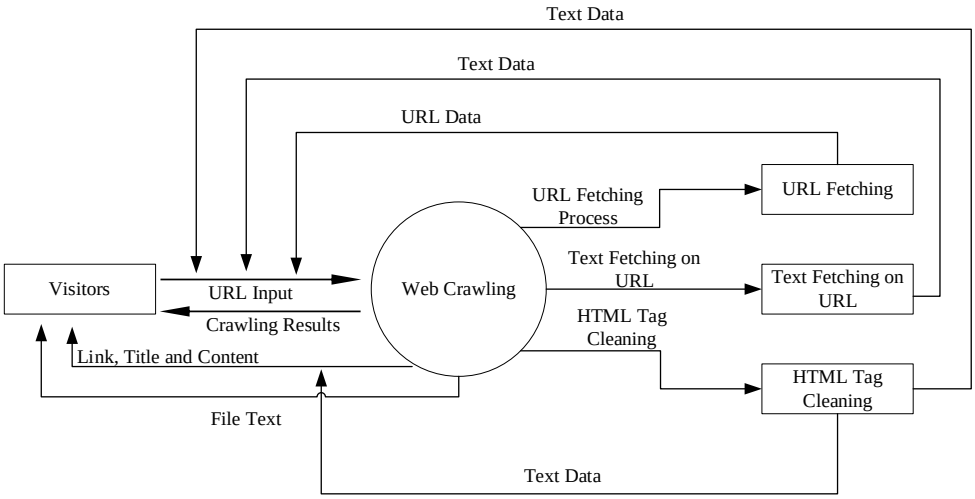


Fig 3. Context Diagram

Description:
1. Visitors enter the URL to be crawled, and then the system will display the crawling results in the form of links, titles, and content
2. When the system crawls the inputted URL, the system fetches the URL on a website.
3. After the URL has been obtained, the system fetches the text on the URL.
4. After the text is obtained, the system cleans the HTML tags on the website so that the final result of the system is a text file.

In this DFD Level 0, the data flow in the system will be described to build a structured search system. The following is a description of the data flow in the crawler system.

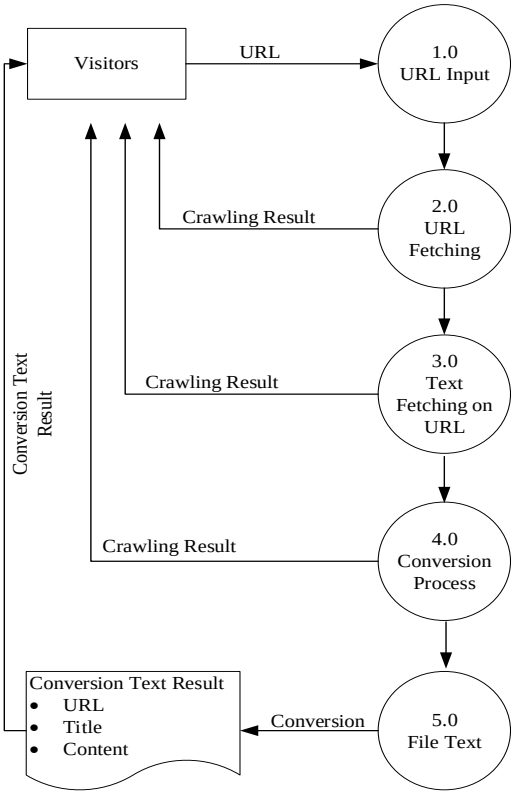


Fig 4. DFD Level 0

Description:
1. Process 1.0 is the process where visitors input the URL.
2. Process 2.0 is the process of taking the URL on a website.
3. Process 3.0 takes text from the URL, and then the system cleans the HTML tags on a website from the text.
4. Process 4.0 is the conversion process or the process of downloading links and text that have been crawled in the form of links, titles, and news content.
5. Process 5.0 is the final process where visitors will get the conversion results as a .txt file.

The user will input the URL that the system wants to crawl. After the URL is entered, the system will automatically read and open the files contained in the targeted website. From the results of this processing, the system then displays a list of URLs that have been

successfully crawled, providing a complete picture of the link structure on the website. This process makes it easier for users to see and access all URLs available on a site.

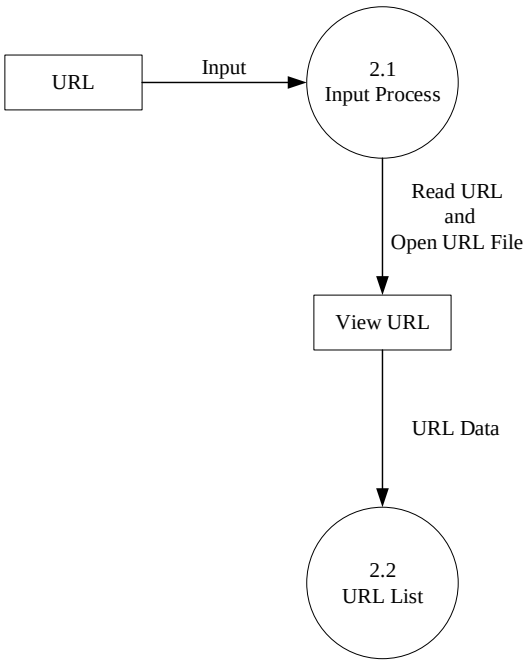


Fig 5. DFD Level 1 Process 2.0 URL Retrieval Process

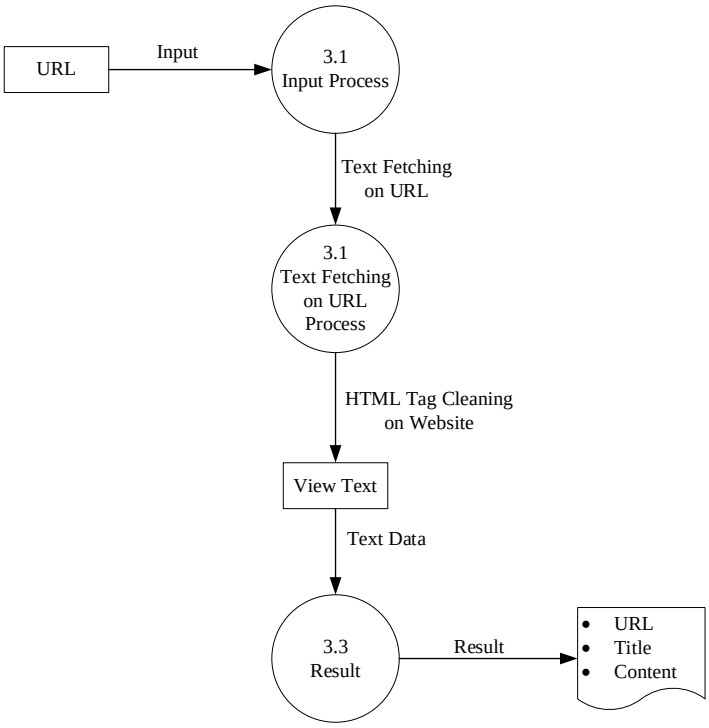


Fig 6. DFD level 1 Process 2.0 Website Analysis Process

Description:

1. Visitors input a URL to be crawled
2. After the input is complete, the system will take the text from the URL and remove all HTML tags on the website. After completion, the system will display the results through links, titles, and content.

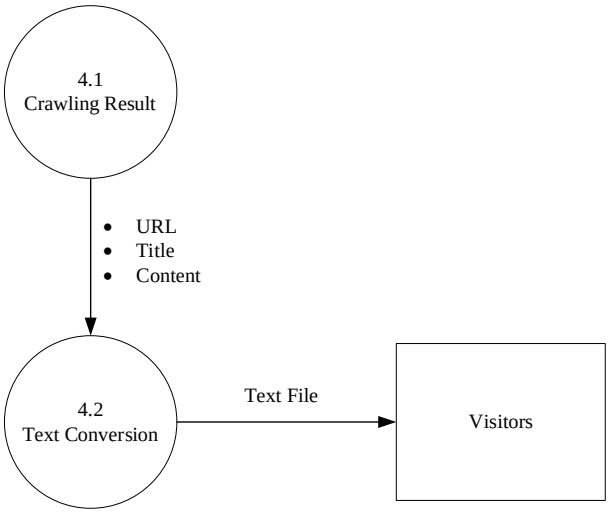


Fig 8. DFD level 1 Process 4.0 Crawling Results

Description: In Figure 8 above, the crawling results in URLs, titles, and content can be converted or downloaded by visitors to save the text file.

The pseudocode of the web crawler is as follows:

```
Procedure Crawler(frontier)
While not frontier.done() do
      Website ← frontier.nextSite()
      url ← website.nextURL()
      if website permitsCrawl(url) then
            text ← retrieveURL.(url)
            storeDocument(url.text)
            end if
            frontier.releaseSite(website)
      end while
end procedure
```

From the pseudocode above, it can be seen that the beginning of the web crawler process is the Crawler Procedure (frontier). The meaning of frontier is a container containing a set of URLs corresponding to the main web page. In a while, not frontier. Done (), it is explained that if the frontier has not finished crawling, the system will search for the following website; this function is described in the syntax Website frontier, following site (). After the system searches for the following website, the system will select several URLs related to the main website; this process is explained in the syntax URL website.nextURL ().
If the website permits crawl (url), then if the website permits crawling the URL, the web crawler will download or call the URL referred to from the main website. Then, the storeDocument(url.text) syntax explains that the system will accommodate the URL and text separated in this syntax frontier.releaseSite(website) explains that after the system is successfully run, the system will display a user interface in the form of a title, URL, and text.

5. Conclusion

This research produces a web crawler application developed using the PHP programming language for indexing text data. This system uses Data Flow Diagram (DFG) modeling to describe the data flow in the application. This application uses the html. domparser code to retrieve all web files from the system, thus enabling comprehensive data retrieval from the website. The test results show that the application successfully stores content by downloading files in the user interface. This system can also retrieve information such as URLs, titles, and content from crawled websites, indicating that the application functions well in indexing web data.

References

[1] F. Imene and J. Imhanzenobe, "Information technology and the accountant today: What has really changed?," *J. Account. Tax.*, vol. 12, no. 1, pp. 48–60, 2020.
[2] D. C. Prakash, R. C. Narayanan, N. Ganesh, M. Ramachandran, S. Chinnasami, and R. Rajeshwari, "A study on image processing with data analysis," in *AIP conference proceedings*, 2022.
[3] M. T. M. Talavera, N. P. Gordoncillo, N. A. Tandang, and D. G. C. Domingo, "Acceptability of height measuring equipment of different materials among community nutrition and health workers and parents in Laguna province, Philippines," *Acta Med. Philipp.*, vol. 56, no. 3, 2022.
[4] L. Rumbo-Rodr\'\iguez, M. Sánchez-SanSegundo, R. Ferrer-Cascales, N. Garc\'\ia-D'Urso, J. A. Hurtado-Sánchez, and A.

Zaragoza-Mart\'\i, "Comparison of body scanner and manual anthropometric measurements of body shape: a systematic review," *Int. J. Environ. Res. Public Health*, vol. 18, no. 12, p. 6213, 2021.
[5] W. Burger and M. J. Burge, *Digital image processing: An algorithmic introduction*. Springer Nature, 2022.
[6] M. Jankowski, D. Gündüz, and K. Mikolajczyk, "Wireless image retrieval at the edge," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 1, pp. 89–100, 2020.
[7] Y. Amit, P. Felzenszwalb, and R. Girshick, "Object detection," in *Computer Vision: A Reference Guide*, Springer, 2021, pp. 875–883.
[8] D. Lee, J. Kim, S. C. Jeong, and S. Kwon, "Human height estimation by color deep learning and depth 3D conversion," *Appl. Sci.*, vol. 10, no. 16, p. 5531, 2020.
[9] B. Dorjee, B. Bogin, C. Scheffler, D. Groth, J. Sen, and M. Hermanussen, "Association of anthropometric indices of nutritional status with growth in height among Limboo children of Sikkim, India," *Anthr. Anz*, vol. 77, pp. 389–398, 2020.
[10] M. J. Hautus, N. A. Macmillan, and C. D. Creelman, *Detection theory: A user's guide*. Routledge, 2021.
[11] K. Williamson, D. N. Blane, and M. E. J. Lean, "Challenges in obtaining accurate anthropometric measures for adults with severe obesity: A community-based study," *Scand. J. Public Health*, vol. 51, no. 6, pp. 935–943, 2023.
[12] C. Morikawa *et al.*, "Image and video processing on mobile devices: a survey," *Vis. Comput.*, vol. 37, no. 12, pp. 2931–2949, 2021.
[13] J. A. Richards, J. A. Richards, and others, *Remote sensing digital image analysis*, vol. 5. Springer, 2022.
[14] K. Lehn, M. Gotzes, and F. Klawonn, "Greyscale and Colour Representation," in *Introduction to Computer Graphics: Using OpenGL and Java*, Springer, 2023, pp. 193–210.
[15] D. Savić, "From Digitization and Digitalization to Digital Transformation: A Case for Grey Literature Management.," *Grey J.*, vol. 16, no. 1, 2020.
[16] R. Thakur and R. Rohilla, "Recent advances in digital image manipulation detection techniques: A brief review," *Forensic Sci. Int.*, vol. 312, p. 110311, 2020.
[17] B. Meyzia, M. Hamdi, R. Amelia, and others, "Imaging analysis of thresholding image filtering, brain abnormalities morphology, and dose report CT scan records," in *Journal of Physics: Conference Series*, 2020, p. 12155.
[18] M. Eminagaoglu, "A new similarity measure for vector space models in text classification and information retrieval," *J. Inf. Sci.*, vol. 48, no. 4, pp. 463–476, 2022.
[19] Z. E. Chay, C. H. Lee, K. C. Lee, J. S. H. Oon, and M. H. T. Ling, "Russel and Rao coefficient is a suitable substitute for Dice coefficient in studying restriction mapped genetic distances of Escherichia coli," *arXiv Prepr. arXiv2302.12714*, 2023.
[20] M. A. de Albuquerque, E. R. do Nascimento, K. N. N. de Oliveira Barros, and P. S. N. Barros, "Comparison between similarity coefficients with application in forest sciences," *Res. Soc. Dev.*, vol. 11, no. 2, pp. e48511226046--e48511226046, 2022.