

C O M M U N I C A T I O N S

FACULTY OF SCIENCES
UNIVERSITY OF ANKARA

DE LA FACULTE DES SCIENCES
DE L'UNIVERSITE D'ANKARA

Series A2-A3: Physical Sciences and Engineering

VOLUME: 65

Number: 1

YEAR: 2023

Faculty of Sciences, Ankara University
06100 Beşevler, Ankara –Türkiye
ISSN 1303-6009 E-ISSN 2618-6462

C O M M U N I C A T I O N S

FACULTY OF SCIENCES
UNIVERSITY OF ANKARA

DE LA FACULTE DES SCIENCES
DE L'UNIVERSITE D'ANKARA

Series A2-A3: Physical Sciences and Engineering

Volume: 65

Number: 1

Year: 2023

Owner

Sait HALICIOĞLU, Dean of Faculty of Sciences

Editor in Chief

Fatma KARAKOÇ, Ankara University

Managing Editor

Şengül KURU, Ankara University

Area Editors

İnanç ŞAHİN (Physics) Ankara University, Türkiye	İman ASKERZADE (Askerbeyli) (Computer Eng.) Ankara University, Türkiye
Handan OLĞAR (Engineering Physics) Ankara University, Türkiye	Ziya TELATAR (Electronic Eng.) Başkent University, Türkiye
H. Volkan ŞENAVCI (Astronomy) Ankara University, Türkiye	M. Emin CANDANSAYAR (Geophysical Eng.) Ankara University, Türkiye

Editors

Ramiz ALIGULIYEV ANAS, Azerbaijan Murat EFE Ankara University, Türkiye Mustafa E. KAMASAK Istanbul Technical University, Türkiye Javier NEGRO Univesidad de Valladolid, Spain Emre YENGEL King Abdullah Uni. of Sci.Tech., Saudi Arabia	Gabriela CIUPRINA Politehnica Univeristy of Bucharest, Romania Osman EROGLU TOBB Economy and Tech. Uni., Türkiye İlhan KOSALAY Ankara University, Türkiye Miroslav VOZNAK VŠB – Tech.Uni. of Ostrava, Czech Republic A. Egemen YILMAZ Ankara University, Türkiye	Sara CRUZ Y CRUZ SEPI-UPIITA IPN, Mexico H. Gokhan İLK Ankara University, Türkiye İsa NAVRUZ Ankara University, Türkiye Hakan TORA Atılım University, Türkiye Kutluay YUCE Ankara University, Türkiye
--	---	--

Language Editor

Fatma KARAKOÇ, Ankara University

Production Editor

Şengül KURU, Ankara University

This Journal is published two issues in a year by the Faculty of Sciences, University of Ankara. Articles and any other material published in this journal represent the opinions of the author(s) and should not be construed to reflect the opinions of the Editor(s) and the Publisher(s).

Correspondence Address:

COMMUNICATIONS
EDITORIAL OFFICE
Ankara University, Faculty of Sciences,
06100 Beşevler, ANKARA – TÜRKİYE
Tel: (90) 312-216 87 00 Fax: (90) 312-216 89 00
e-mail: commun@science.ankara.edu.tr
<http://communications.science.ankara.edu.tr/index.php?series=A2A3>

Print:

Ankara University Press
İncitaş Sokak No:10 06510 Beşevler
ANKARA – TÜRKİYE

C O M M U N I C A T I O N S

FACULTY OF SCIENCES
UNIVERSITY OF ANKARA

DE LA FACULTE DES SCIENCES
DE L'UNIVERSITE D'ANKARA

Series A2-A3: Physical Sciences and Engineering

Volume: 65

Number: 1

Year: 2023

Research Articles

Ö.F. YANIK, H.A. ILGIN, A comprehensive computational cost analysis for state-of-the-art visual slam methods for autonomous mapping.....	1
K. AÇICI, Ö.M. ERDAL, A. YILMAZ, M. UNAL, G.E. BOSTANCI, M.S. GÜZEL, Airborne imagery and lidar based 3D reconstruction using commercial drones.....	16
F. EKİNCİ, Ionization and phonon production by ¹⁰ B ions in radiotherapy applications.....	30
K. GÜRCÜOĞLU, T.D. MEDENİ, İ.T. MEDENİ, M.S. GÜZEL, H. ARSLAN, Ontology development for web services to be used within the scope of remote monitoring project.....	38
M. ÇOLAK, T.T. SİVRİ, N.P. AKMAN, A. BERKOL, Y. EKİCİ, Disease prognosis using machine learning algorithms based on new clinical dataset	52
İ. KARATEPE, V. NABIYEV, Military camouflage classification with Mask R-CNN algorithm.....	69

A COMPREHENSIVE COMPUTATIONAL COST ANALYSIS FOR STATE-OF-THE-ART VISUAL SLAM METHODS FOR AUTONOMOUS MAPPING

Ömer Faruk YANIK¹ and Hakkı Alparslan ILGIN¹

¹Department of Electrical and Electronics Engineering,
Ankara University, Ankara, TÜRKİYE

ABSTRACT. It is important to solve the autonomous mapping problem with high accuracy using limited energy resources in an environment without prior knowledge and/or signal. Visual Simultaneous Localization and Mapping (SLAM) deals with the problem of determining the position and orientation of an autonomous vehicle or robot with various on-board sensors, and simultaneously creating a map of environment with low energy consumption. However visual SLAM methods require high processing performance for real-time operations. Also, processing capability of the hardware is limited by the power constraints. Therefore, it is necessary to compare the processing load and power consumption of visual SLAM methods for autonomous vehicles or robots. For visual SLAM methods, although there are different comparison studies, there is no comprehensive computational cost analysis covering different datasets and important parameters including absolute trajectory error, RAM Usage, CPU load, GPU load, with total power consumption. In this paper, ORB-SLAM2, Direct Sparse Odometry (DSO), and DSO with Loop Closure (LDSO), which are state of the art visual SLAM methods, are compared. Besides the performance of these methods, energy consumption and resource usage are evaluated allowing the selection of the appropriate SLAM method.

1. INTRODUCTION

Autonomous systems have become more visible in daily life, with many products from driverless cars to cleaning robots, from armed unmanned aerial vehicles to consumer electronics. They carry out specific missions autonomously with limited sources providing convenience and assistance to humans.

Keywords. Localization, mapping, indoor navigation, visual SLAM, computational cost.

✉ ofyanik@ankara.edu.tr;  0000-0001-7832-1945

✉ ilgin@eng.ankara.edu.tr-Corresponding author;  0000-0003-0112-4833.

In environments where previously known or satellite signals can be received, relatively uncomplicated autonomous movement capability such as waypoint identification can be achieved with conventional applications. Position information can be obtained through Global Positioning System (GPS) receiver if satellite signal is available. In addition, various devices and sensors are also used to solve the same problem. Among them, Light Detection and Ranging (LIDAR) is used to determine position of the robot. Inertial Measurement Unit (IMU) is another device with accelerometer and gyroscope that is used widely. Position information can be used in a certain standard for functions such as waypoint identification, lane tracking, estimation of the distance passed, obstacle avoidance and distance adjustment. In these applications, the robot, whose spatial relationship is known through its sensors, estimates its own position by using the relevant reference information. However, in an environment where GPS satellite signals cannot be detected by the robot or in a place for which no prior knowledge is available, these methods, due to the high margin of error, are not sufficient in terms of functions such as obstacle avoidance and determining momentary position in its surroundings. In environments that are previously unknown or where GPS signals cannot be detected, different solutions are needed to calculate the robot's position relative to the environment.

There are many studies that include various methods such as motion estimation, target tracking, waypoint tracking based on GPS and different interpretations based on these methods in order to increase autonomous movement capability. However, in order to determine the location of a robot in environments without prior knowledge, it is necessary to obtain information about the environment first and to calculate the position of the robot while obtaining information about the environment.

1.1. Definition of SLAM Problem. If there is no information about the surroundings of wheeled robots, measurement of distances by odometer can be used. However, wheel slipping or spinning errors accumulate and increase the total error in the wheel odometry, as the error in the previous position information will also affect the next position information. Additionally, this does not apply to wheelless vehicles and robots. For this reason, different methods based on laser and visual sensors are used to obtain location information. Among these methods, odometry is based on calculating the current position with respect to previously visited locations to follow the course of the robot. On the contrary, for SLAM methods, position and environment are considered together in processing [1].

SLAM is the method that deals with the computational problem of tracking a robot's position while simultaneously creating or updating a map of a previously unknown environment. Since robot has to do two related tasks at the same time, the SLAM problem is a complex issue that needs to be solved in real-time [2].

Calculation of displacement performed by the robot (odometry) and understanding of a place passed again (loop closure) are the basic components of SLAM [3]. In addition to being formulated in many theoretical ways, SLAM has the opportunity to be applied in many areas from indoor robots to underwater and air systems. The solution to the SLAM problem, which enables robots to be truly autonomous, is seen as one of the most significant achievements of robotics field [4].

As well as there are various methods to achieve autonomous mapping solution, there also exists data utilized to evaluate and compare the performance of those methods. Some of the data are obtained in the controlled environments using different techniques. For evaluation in different conditions, there exist different data with specific properties. Thus, it is possible to perform performance tests for visual SLAM methods in various conditions such as surroundings with moving objects. In the next subsections, information about the probabilistic definition of SLAM, environmental factors, and related works are given.

1.2. Solution of SLAM Problem. The position of a robot can be calculated relatively easily if the robot has a precise prior knowledge of the environment. On the contrary, a robust model of the environment in which the robot is located can be created if the position of the robot is known perfectly. However, within the scope of SLAM problem, the robot with on-board sensors does not have any prior knowledge about the environment it is located in. Probabilistic methods are used in solving the SLAM problem, since the robot's location and environment's map are created simultaneously and a perfect information cannot be obtained [3]. In SLAM problem (see Fig. 1), if \mathbf{x}_k is the robot's position and orientation, and \mathbf{u}_k is the control input applied to bring the robot to position \mathbf{x}_k at time $k - 1$, \mathbf{m}_i is the location of each obstacle that makes up the environment map; \mathbf{z}_{ik} refers to observation of the robot at time k .

Also, $\mathbf{X}_{0:k} = \{\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_k\}$ are the positions where the robot has visited in the past. $\mathbf{U}_{1:k} = \{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_k\}$ are history of control inputs. Robot's all observations on landmarks are $\mathbf{Z}_{0:k} = \{\mathbf{z}_0, \mathbf{z}_1, \dots, \mathbf{z}_k\}$. Together with these definitions, solution of the SLAM problem is based on the probabilistic approach given below as conditional probability (\mathbf{P}), which must be calculated across whole trajectory:

$$\mathbf{P}(\mathbf{x}_k, \mathbf{m} | \mathbf{Z}_{0:k}, \mathbf{U}_{1:k}, \mathbf{x}_0) \quad (1)$$

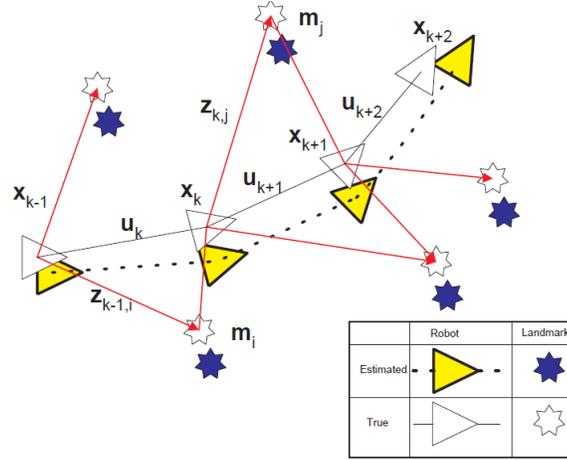


FIGURE 1. Solution of SLAM problem [4].

With the probabilistic definition given in equation (1), many methods have emerged for solution of the SLAM problem. In order to perform these solutions in real-time, extended Kalman filter, particle filter and graph-based methods are mostly used [5].

Besides theoretical solution, autonomous systems in daily life are expected to be able to cope with the usual environmental factors. However, there are many compelling factors within the scope of solving the SLAM problem. Effects such as sensor noise, size of the area where the robot is located, and the dynamic elements in the environment cause mathematical and hardware difficulties in solving SLAM problem. The accuracy of the sensors, as an indicator of the reliability of the data regarding environment and the state of the robot, emerges as an important factor for solution of the SLAM problem. In addition, size of the environment reveals as another problem, as size of the environment will increase the need for memory, processing load, and accumulation of error originating from the sensor. Moving objects, which change model of the environment, are another environmental factor that should be taken into account for solution of the SLAM problem.

In addition to the variety of SLAM's mathematical solution methods, different devices such as camera, accelerometer are used in robots. Robustness of the sensor data will reduce the error accumulated. However, since it is not possible to obtain perfect sensor data in real life, the solution to the SLAM problem should be able to overcome sensor errors and noise. In solution of the SLAM problem, information can be obtained from a single sensor, as well as environmental data can be obtained via multiple sensors of the same type or combinations of different sensors.

1.3. Visual SLAM. SLAM methods using camera as sensor are called visual SLAM and are generally divided into two categories, as direct methods that using entire image and the methods utilizing features [6]. To date, many studies have been carried out in single-camera and multi-camera sensor configurations. Among these, as given in Tab. 1, feature-based PTAM [7], ORB-SLAM [8] created by using ORB features developed over [9] PTAM; direct methods, Large Scale Direct Monocular (LSD) SLAM [10], SVO [11], Direct Sparse Odometry (DSO), Direct Sparse Odometry with Loop Closing (LDSO), Dense Visual (DVO) SLAM [12], and ElasticFusion [13] can be counted. There are also many benchmarks used for performance analysis of the visual SLAM problem. These benchmarks are mostly obtained by capturing movement of objects in controlled environments [14] and are separated from each other by several features like outdoor, indoor including moving objects and/or relatively stationary environments. KITTI, EUROC, TUM RGB-D [14], TUM Monocular [15], ICL-NUIM, Sintel [16], Tsukuba [17] and NYU [18] datasets are among them. RAM and CPU usages with processing time on a particular dataset are also compared [19] and for MSCKF [20], OKVIS [21], ROVIO [22], VINS-Mono [23], SVO [11] + MSF [24] and SVO + GTSAM [25]. These studies carried out on four different hardware with Intel and ARM architectures. ATE parameter results are evaluated using some SLAM methods on a wheeled robot with a camera [6]. NVIDIA Jetson TX1 hardware with Ubuntu 16.04 version was used on the robot. LSD SLAM, ORB-SLAM and Direct Sparse Odometry (DSO) were compared. Among the stereo-camera methods, Real-Time Appearance-Based Mapping (RTAB map) [26], ORB SLAM, Stereo Parallel Tracking and Mapping (SPTAM) [27] were compared. ORB2, DynaSLAM [28] and DSO algorithms are also benchmarked [29]. Intel processors were used as part of the analysis. Several parameters including Absolute Trajectory Error were evaluated on TUM Monocular and EUROC data sets.

Although there are many comparison studies, no comprehensive computational cost analysis for visual SLAM methods has been encountered in the literature, apart from comparison of CPU and memory usages for visual-Inertial Navigation methods [19].

TABLE 1. Visual SLAM algorithms.

Algorithm	Method	Map Sparsity	Loop Closing
Parallel Tracking and Mapping (PTAM)	Feature-based	Yes	No
Semi-direct Visual Odometry	Semi-Direct	Yes	No
ORB-SLAM	Feature-based	Semi	Yes
Direct Sparse Odometry (DSO)	Direct	No	No
DSO with Loop Closing	Direct	No	Yes
LSD-SLAM	Direct	No	Yes
DVO-SLAM	Direct	No	Yes
ElasticFusion	Direct	No	Yes

We evaluate performances of state-of-the-art methods such as ORB-SLAM2 (or ORB2) [30], DSO [31], and LDSO [32] on datasets created in different laboratories and separated from each other in terms of information such as indoor and outdoor environments. The most important parameters, which are Central Processing Unit (CPU), Graphics Processing Unit (GPU), and Random Access Memory (RAM) usage as well as power consumption of the SLAM algorithms are compared. On the other hand, related to the performance of visual SLAM methods, this study represents a different and comprehensive perspective using various datasets such as ICL-NUIM, KITTI, EUROCC and TUM Monocular, regarding to hardware constraints such as power consumption, memory usage, CPU and GPU load. Material and Methods, Experimental Results, and Conclusion about the study are given in the next three sections.

2. MATERIALS AND METHODS

In this paper, performance and behaviours of ORB2, DSO, and LDSO algorithms on KITTI, EUROCC, TUM Monocular and ICL-NUIM datasets were compared using NVIDIA Jetson TX1 hardware in real-time (online). ATE parameter was obtained via “evo” repository [33] to detect trajectory error and to get graphical results. In order for the datasets to be used by the visual SLAM methods, calibration of the data, definition of the timestep of each image, correct naming of the images, method for reading timesteps are considered. According to SLAM methods, data set formats are also tailored. Analyses were performed on NVIDIA Jetson TX1 hardware which has NVIDIA Maxwell GPU (with 256 NVIDIA CUDA Cores). It has also ARM Cortex Quad Core CPU, 4 GB LPDDR memory with Ubuntu 18.04 operating system. In order to determine the hardware parameters, “tegrastats” software interface provided by NVIDIA JETSON TX1 was used. “tegrastats” interface provides a text file containing detailed information about resources such as RAM usage, GPU and CPU loads and total consumed power in desired period. Tegrastats data obtained at frequency of 1 Hz were converted to .mat file via MATLAB.

2.1. SLAM Methods Used in Benchmarks. ORB2 and Direct Sparse Odometry with and without loop closing algorithms are compared in the experimental studies. ORB2, a feature-based SLAM method, can produce solutions with monocular, stereo and RGB-D cameras. On the other hand, LDSO and DSO methods produce monocular solutions. Therefore, monocular solutions were emphasized in the comparisons. ORB2 has map reuse, loop closure and relocalization capabilities. ORB2 can be executed in real-time for synthetic environments as well as indoor and outdoor sequences obtained by cars or robots. It uses ORB features for mapping, tracking, relocalization and loop closing [30].

DSO is a monocular visual odometry method, which can be executed in real-time. DSO combines a full photometric calibration by using lens vignetting, exposure time, and non-linear responses. Without need for features, it is able to sample pixels through the image areas which have density gradient [31].

LDSO method was created using the point selection infrastructure of the DSO method. But it has loop closure feature that enables detection of repeated point. In this respect, LDSO is a monocular Visual SLAM method [32].

2.2. Benchmark Datasets. In order to compare SLAM methods, datasets developed in laboratory environments or developed with special equipment are used. There are many datasets available from open sources differentiated from each other according to the environments they describe, and the equipment they use to obtain images. As such, using datasets with different characteristics will be an important approach to reach an accurate result. For this reason, within the scope of this study, KITTI, ICL-NUIM, TUM Monocular and EUROC benchmark datasets, which are available in open sources, were used to compare SLAM methods.

KITTI outdoor data set was captured using various sensors. Therefore, the data set includes images captured by camera beside the measurements obtained via GPS and position information. Also, accelerometer data is supplied [34]. KITTI benchmark provide 11 training and 11 test data, which can also be used with stereo methods (Visual Odometry / SLAM Evaluation, 2012). KITTI data sets named 00, 03 and 07 each with 10 frame per second were used in the comparison of the algorithms ORB2, DSO and LDSO. A micro unmanned aerial vehicle equipped with stereo camera was used to create EUROC data set, which is available publicly. Beside the image data there are also measurement results by accelerator added to the same data set. This data set includes three categories according to degree of difficulty. In the experiments data with various difficulty degrees from this set, which are MH01, V102 and V203 from interior space were used, so that the performance and consumption values of the algorithms are evaluated extensively. For data that is rated from easy to difficult, motion and thus blur increases progressively. ICL-NUIM dataset contains RGB-D data using handheld camera movements [35]. Unlike other datasets, images in the ICL-NUIM dataset were not collected from a real environment. The ICL-NUIM dataset consists of synthetic images and depth information with frequency of 30 Hz, and ground-truth data. ICL-NUIM dataset is divided into two categories as Office Room and Living Room, each with four subsets. It was considered that LivingRoom0 (lR0) and OfficeRoom0 (oR0) which have the maximum number of video frames among the all subsets are appropriate to be used in the analysis. TUM Monocular dataset contains fisheye camera video frames from small indoor to large outdoor environments. It has also ground truth data, and calibration parameters in 50 subsets. Subsets numbered 19, 29, and 30 are used for comparison of LDSO and DSO algorithms. Because ORB2

is not suitable with fisheye camera frame, ORB2 was not compared on TUM Monocular dataset.

3. EXPERIMENTAL RESULTS

In experimental studies, video sequences from TUM Monocular, EUROCC, KITTI and ICL-NUIM data sets are used for benchmarking ORB2, LDS and DSO algorithms. Total of five parameters, which are ATE, RAM usage, CPU and GPU loads and total power consumption are compared in the experiments. In order to avoid non-deterministic results, each SLAM method was executed six times on the video sequences. At the end, average of six results obtained for the parameters was used for the comparisons. Details of the results obtained for the parameters are given in the following subsections.

In order to obtain data that is in-use by RAM, GPU Load, CPU Load, and Consumed Total Power, “Tegrastats” software is used. ATE results are obtained by using Evo software. Evo software supplies ATE parameter as meter (rms). By means of Tegrastats, RAM usage is obtained as Megabyte (MB). GPU Load and Consumed Total Power are determined in term of milliwatt. CPU load is measured as percentage.

3.1. Absolute Trajectory Error. ATE is one of the most frequently used parameters in the comparison of SLAM methods. It is an indicator of how well the SLAM method can track ground-truth data. For the calculation of ATE parameter, open-source Evo tool was used, and the trajectory file was compared with the ground-truth data. Results obtained for ICL-NUIM, KITTI, EUROCC and TUM Monocular datasets are given in Tab. 2. DSO method gives the best result on OfficeRoom0, while ORB2 gives the best result on all other datasets. Besides, LDSO is more accurate than DSO for TUM Monocular dataset. Trajectory graph of OfficeRoom0 is given in Fig. 2, in which all methods track the ground-truth similar to each other with tolerable differences.

3.2. RAM Usage. RAM usage is also an important parameter, which is used to quantify performance of the algorithms. It is aimed to examine RAM load in order to evaluate how SLAM methods use hardware. Therefore, RAM loads created by ORB2, LDSO DSO methods over the video sequences of EUROCC, ICL-NUIM, KITTI, and TUM data sets were recorded with the frequency of 1 Hz. RAM usage results are given in Tab. 3. DSO gives the best results in RAM usage due to the lack of loop closure increasing complexity.

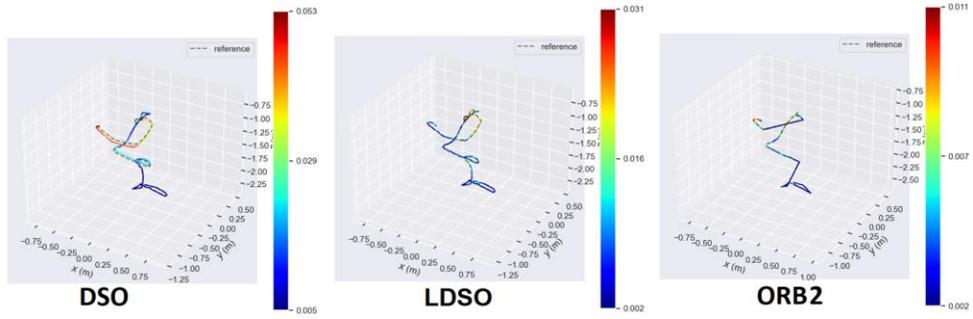


FIGURE 2. Trajectory of methods on officeroom0: DSO (left) gives the best result in terms of ATE.

TABLE 2. ATE results (rms).

	ICL-NUIM		EUROC			KITTI			TUM Monocular		
	oR0	IR0	MH01	V102	V203	00	03	07	19	29	30
DSO	0.162	0.024	0.0464	1.026	1.373	119.77	2.314	15.272	0.165	0.147	0.365
LDSO	0.413	0.114	0.0425	1.511	1.142	10.612	2.904	6.385	0.137	0.057	0.085
ORB2	0.171	0.008	0.0441	0.064	0.263	8.3245	1.807	2.132	-	-	-

TABLE 3. RAM usage - MB (rms).

	ICL-NUIM		EUROC			KITTI			TUM Monocular		
	oR0	IR0	MH01	V102	V203	00	03	07	19	29	30
DSO	1362	1354	1430	1423	1443	1701	1626	1509	1804	1768	1554
LDSO	1974	1980	2177	2121	2143	2943	2112	2071	2429	2402	2019
ORB2	1515	1547	1558	1525	1524	2160	1457	1475	-	-	-

TABLE 4. CPU usage (rms).

	ICL-NUIM		EUROC			KITTI			TUM Monocular		
	oR0	IR0	MH01	V102	V203	00	03	07	19	29	30
DSO	47.292	45.876	45.815	50.562	54.692	61.974	52.661	56.371	57.365	55.167	53.272
LDSO	51.374	49.200	53.796	54.595	55.334	68.162	56.502	59.068	59.114	57.745	57.646
ORB2	55.753	53.813	53.858	52.519	49.349	39.928	46.925	48.067	-	-	-

TABLE 5. Total power consumption – milliwatts (rms).

	ICL-NUIM		EUROC			KITTI			TUM Monocular		
	oR0	IR0	MH01	V102	V203	00	03	07	19	29	30
DSO	5558.8	5368.4	6515.6	6250.9	6727.4	8878.6	6070.0	6547.6	8033.2	7377.7	6382.8
LDSO	5747.6	5541.9	6362.5	5940.1	5935.0	6875.9	5926.4	6323.5	6727.0	6297.7	6185.9
ORB2	5351.8	5236.2	5465.1	5248.5	5201.9	4511.5	4959.4	5022.4	-	-	-

3.3. CPU Usage. One of the parameters frequently used in the computational load calculations of SLAM methods is CPU load. Since NVIDIA Jetson TX1 has quad-core processor, comparisons were made by averaging the usage rates of four processors. CPU load results are given in Table . LDSO method causes higher CPU usage than DSO for all datasets. This result is about loop closure effect. On the other hand, while ORB2, as a feature-based method, causes more CPU load in case of non-textured environment such as officeRoom0, livingRoom0, and MH01, it causes less CPU usage in more textured environment such as V203, 00, and 07.

3.4. Total Power Consumption. One of the most important constraints for autonomous systems is limited power supplies. For this reason, it will be a correct approach to examine how the total power consumption of the hardware changes depending on the SLAM methods, as well as processing load on the CPU. Total power consumption results are given in Table . 5. According to results shown in Table 5, the least power consuming method is ORB2. Furthermore, when compared to DSO, it is obviously seen that LDSO method consumes less power, with an exception of ICL-NUIM dataset on which total power consumption of the methods are similar. This situation is also a question of this study to answer because power consumption behavior is different from the CPU load behavior.

3.5. GPU Power Consumption. The difference between CPU usage and overall power consumed behaviors by the hardware, which is created by SLAM methods has revealed the need to check GPU load. The results of given data provided by hardware have been able to explain the differences between CPU loads and total power dissipated.

GPU load in milliwatts of all methods are given in Tab6. For all datasets, GPU power consumption of DSO is higher than the others while ORB2 is the most effective one among all methods for all datasets. GPU power consumption versus time graphs of all methods are also given in Fig. 3. As seen in this figure, GPU load of DSO method for all dataset is getting higher over time. As explained before, on TUM Monocular Dataset, ORB2 methods was not executed.

TABLE 6. Power consumption by GPU – milliwatts (rms).

	ICL-NUIM		EUROC			KITTI		
	oR0	IR0	MH01	V102	V203	00	03	07
DSO	346,7439	350,9953	1303,57	1052,36	1472,55	3050,55	621,99	1162,24
LDSO	304,4431	341,4259	902,00	576,95	550,76	995,11	520,67	631,24
ORB2	165,4507	148,8809	99,02	120,71	144,56	81,85	110,61	85,52

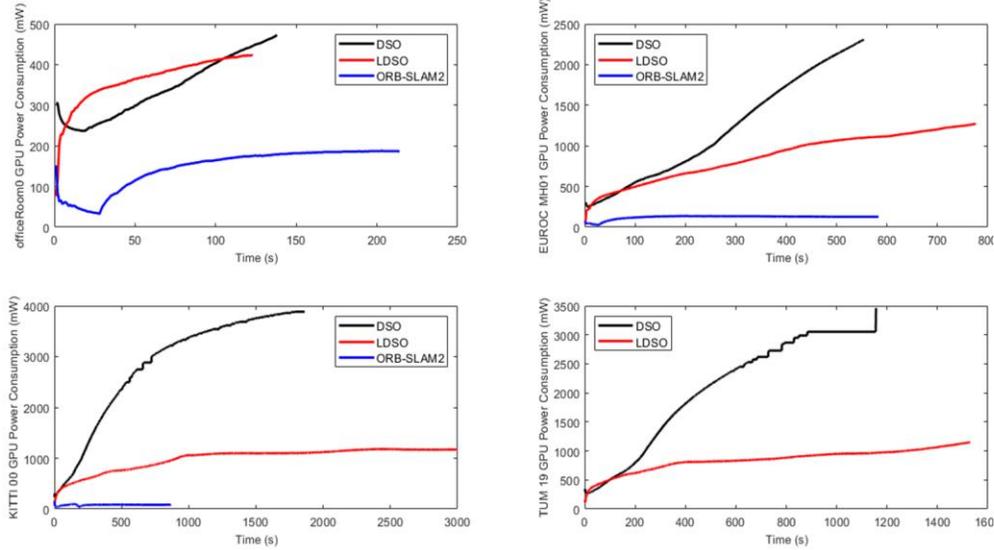


FIGURE 3. GPU power consumption of DSO, LDSO and ORB-SLAM2.

4. CONCLUSION

In this paper, performances of ORB2, DSO and LDSO methods, which are suitable to be executed on ARM processor that NVIDIA Jetson TX1 has, are compared using ICL-NUIM, KITTI, TUM Monocular and EUROC benchmark datasets. By making an exhaustive comparison, in particular, it is aimed to make a choice among the methods in order to meet the power constraints of mobile robots, localization requirements within indoor environment, and to guide to future studies. ATE, RAM usage, CPU and GPU loads and total consumed power parameters were used for the comparison studies.

ORB2 is the most effective method for ATE parameter, except for the Officeroom0 dataset, for which DSO method gives the best result. The reason for ORB2 gives the worst result for OfficeRoom0 is that it cannot detect features in the Officeroom0 dataset sufficiently.

Due to the lack of loop closure in DSO, it gives the best results in RAM usage as expected. On the other hand, as a direct method, owing to loop-closure ability, LDSO is the method that pushes RAM constraints the most. It has been determined that the LDSO method not only forces the RAM constraint of NVIDIA Jetson TX1 hardware during the process, but also tends to use the SWAP memory more.

As for the CPU load, DSO method gives better results on the ICL-NUIM dataset, while the ORB2 method has better results on the KITTI dataset. However, while DSO method is better for MH01 and V102 dataset, ORB2 method achieves better results on V203 dataset. In this way, it can be said that ORB2 causes more CPU load when solving SLAM problem in surroundings such as supplied by ICL-NUIM where features are less obvious, whereas ORB2 algorithm also causes CPU to have less load in environments where features are dense and can be followed.

When the CPU load and total power consumption are compared, different behaviours are noticed. For instance, CPU is used at least by DSO algorithm on MH01 data set. However, power consumed by this method is the most demanding among all. As an explanation for that, DSO is the method that requires the most GPU power.

For GPU power consumption, the best results were obtained with ORB2 on all datasets while the worst results were obtained by DSO. Differences between GPU power consumption behaviours are related with keyframe frequencies of the methods. DSO and LDSO use 5-10 frames per second [31], and 5-7 frames per second [32], respectively. However, ORB2 uses one frame as a keyframe per 20 frames maximum [30]. This situation is the root cause for the difference between the total power consumption and CPU load.

LDSO method is not suitable to be used in large environments, since it pushed RAM constraints. In textured environments, regardless of the size of the

environment, ORB2 is more suitable than DSO and LDSO to solve the SLAM problem within the constraints of the hardware in terms of ATE, GPU and CPU loads, and total power consumption. On the other hand, within the untextured environment with short and straight trajectory, DSO is suitable for the solution of the SLAM problem.

Author Contribution Statements The authors equally worked on the study. All authors read and approved the final copy of the manuscript.

Declaration of Competing Interests The authors declare that there is no conflict of interest regarding the publication of manuscript.

REFERENCES

- [1] Yousif, K., Bab-Hadiashar, A., Hoseinnezhad, R., An overview to visual odometry and visual SLAM: Applications to mobile robotics, *Intell. Ind. Syst.*, 1 (4) (2015), 289-311, <https://doi.org/10.1007/s40903-015-0032-7>.
- [2] Huletski, A., Kartashov, D., Krinkin, K., Evaluation of the modern visual SLAM methods, *Artificial Intelligence and Natural Language and Information Extraction, Social Media and Web Search FRUCT Conference (AINL-ISMW FRUCT)*, (2015), 19-25, <https://doi.org/10.1109/AINL-ISMW-FRUCT.2015.7382963>.
- [3] Guçlu, O., Simultaneous Localization and Mapping Using RGB-D Sensors, (2018). Doctoral Thesis, Hacettepe University, Turkey.
- [4] Durrant-Whyte, H., Bailey, T., Simultaneous localization and mapping (SLAM): Part I the essential algorithms, *IEEE Robot. Autom. Mag.*, 13 (2) (2006), 99-110, <https://doi.org/10.1109/MRA.2006.1638022>.
- [5] Chong, T. J., Tang, X. J., Leng, C. H., et al., Sensor technologies and simultaneous localization and mapping (SLAM), *Procedia Compt. Sci.*, 76 (2015), 174-179, <https://doi.org/10.1016/j.procs.2015.12.336>.
- [6] Filipenko, M., Afanasyev, I., Comparison of various SLAM systems for mobile robot in an indoor environment, *International Conference on Intelligent Systems (IS)*, (2018), 400-407, <https://doi.org/10.1109/IS.2018.8710464>.
- [7] Klein, G., Murray, D., Parallel tracking and mapping for small AR workspaces, *6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, (2007), 225-234.
- [8] Mur-Artal, R., Montiel, J. M. M., Tardos, J. D., ORB-SLAM: A versatile and accurate monocular SLAM system, *IEEE Trans. Robot.*, 31 (5) (2015), 1147-1163, <https://doi.org/10.1109/TRO.2015.2463671>.
- [9] Liang, X., Chen, H., Li, Y., Liu, Y., Visual laser-SLAM in large-scale indoor environments, *International Conference on Robotics and Biomimetics*, (2016), 19-24.
- [10] Engel, J., Schops, T., Cremers, D., LSD-SLAM: Large-scale direct monocular SLAM,

- European Conference on Computer Vision*, (2014), 834-849, https://doi.org/10.1007/978-3-319-10605-2_54.
- [11] Forster, C., Pizzoli, M., Scaramuzza, D., SVO: Fast semi-direct monocular visual odometry, *IEEE International Conference on Robotics and Automation (ICRA)*, (2014), 15-22, <https://doi.org/10.1109/ICRA.2014.6906584>.
- [12] Kerl, C., Sturm, J., Cremers, D., Dense visual SLAM for RGB-D cameras, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (2013), 2100-2106, <https://doi.org/10.1109/IROS.2013.6696650>.
- [13] Whelan, T., Salas-Moreno, R., Glocker, B., ElasticFusion: Real-time dense SLAM and light source estimation, *Int. J. Robot. Res.*, 35 (14) (2016), 1697-1716, <https://doi.org/10.1177/0278364916669237>.
- [14] Sturm, J., Engelhard, N., Endres, F., A benchmark for the evaluation of RGB-D SLAM systems, *International Conference on Intelligent Robots and Systems*, (2012), 573-580, <https://doi.org/10.1109/IROS.2012.6385773>.
- [15] Engel, J., Usenko, V., Cremers, D., A photometrically calibrated benchmark for monocular visual odometry, (2016), arXiv:1607.02555.
- [16] Butler, D. J., Wulff, J., Stanley, G. B., A naturalistic open source movie for optical flow evaluation, *European Conference on Computer Vision*, (2012), 611-625, https://doi.org/10.1007/978-3-642-33783-3_44.
- [17] Peris, M., Martull, S., Maki, A., Towards a simulation driven stereo vision system, *Proceedings of the 21st International Conference on Pattern Recognition*, (2012), 1038-1042.
- [18] Silberman, N., Hoiem, D., Kohli, P., Indoor segmentation and support inference from RGBD images, *European Conference on Computer Vision*, (2012), 746-760.
- [19] Delmerico, J., Scaramuzza, D., A benchmark comparison of monocular visual-inertial odometry algorithms for flying, *International Conference on Robotics and Automation (ICRA)*, (2018), 2502-2509, <https://doi.org/10.1109/ICRA.2018.8460664>.
- [20] Mourikis, A. I., Roumeliotis, S. I., A multi-state constraint Kalman filter for vision-aided inertial navigation, *Proceedings IEEE International Conference on Robotics and Automation*, (2007), 3565-3572, <https://doi.org/10.1109/ROBOT.2007.364024>.
- [21] Leutenegger, S., Lynen, S., Bosse, M., Keyframe-based visual-inertial odometry using nonlinear optimization, *Int. J. Robot. Res.* 34 (3) (2015), 314-334, <https://doi.org/10.1177/0278364914554813>.
- [22] Bloesch, M., Omari, S., Hutter, M., Robust visual inertial odometry using a direct EKF-based approach, *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (2015), 298-304, <https://doi.org/10.1109/IROS.2015.7353389>.
- [23] Qin, T., Li, P., Shen, S., Vins-mono: A robust and versatile monocular visual-inertial state estimator, *IEEE Transactions on Robotics*, 34 (4) (2018), 1004-1020,

- <https://doi.org/10.1109/TRO.2018.2853729>.
- [24] Lynen, S., Achtelik, M. W., Weiss, S., A robust and modular multi-sensor fusion approach applied to MAV navigation, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (2013), 3923-3929, <https://doi.org/10.1109/IROS.2013.6696917>.
- [25] Forster, C., Carlone, L., Dellaert, F., On-manifold preintegration for real-time visual-inertial odometry, *IEEE Trans. Robot.*, 33 (1) (2016), 1-21, <https://doi.org/10.1109/TRO.2016.2597321>.
- [26] Labbe, M., Michaud, F., Online global loop closure detection for large-scale multi-session graph-based SLAM, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (2014), 2661-2666, <https://doi.org/10.1109/IROS.2014.6942926>.
- [27] Pire, T., Fischer, T., Castro, G., S-PTAM: Stereo parallel tracking and mapping. *Robotics and Autonomous Systems*, 93 (2017), 27-42, <https://doi.org/10.1016/j.robot.2017.03.019>.
- [28] Bescos, B., Fàcil, J. M., Civera, J., DynaSLAM: Tracking, mapping, and inpainting in dynamic scenes, *IEEE Robot. Aut. Let.*, 3 (4) (2018), 4076-4083, <https://doi.org/10.1109/LRA.2018.2860039>.
- [29] Mingachev, E., Lavrenov, R., Tsoy, T., Comparison of ROS-based monocular visual SLAM methods: DSO, LDSO, ORB-SLAM2 and DynaSLAM, *International Conference on Interactive Collaborative Robotics*, (2020), 222-233, https://doi.org/10.1007/978-3-030-60337-3_22.
- [30] Mur-Artal, R., Tardós, J. D., ORB-SLAM2: An open-source slam system for monocular, stereo, and RGB-D cameras, *IEEE Trans. Robot.*, 33 (5) (2017), 1255-1262, <https://doi.org/10.1109/TRO.2017.2705103>.
- [31] Engel, J., Koltun, V., Cremers, D. Direct sparse odometry, *IEEE Trans. Pattern Anal. Mach. Intell.*, 40 (3) (2018), 611-625, <https://doi.org/10.1109/TPAMI.2017.2658577>.
- [32] Gao, X., Wang, R., Demmel, N., Cremers, D., LDSO: Direct sparse odometry with loop closure, *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (2018), 2198-2204.
- [33] Grupp, M., evo: Python package for the evaluation of odometry and SLAM, (2021). Available at: <https://github.com/MichaelGrupp/evo>.
- [34] Geiger, A., Lenz, P., Stiller, C., Vision meets robotics: The KITTI dataset, *Int. J. Robot. Res.*, 32 (11) (2013), 1231-1237, <https://doi.org/10.1177/0278364913491297>.
- [35] Handa, A., Whelan, T., McDonald, J., A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM, *IEEE International Conference on Robotics and Automation (ICRA)*, (2014), 1524-1531, <https://doi.org/10.1109/ICRA.2014.6907054>.

AIRBORNE IMAGERY and LIDAR BASED 3D RECONSTRUCTION USING COMMERCIAL DRONES

Koray AÇICI¹, Ömer Mert ERDAL², Alperen YILMAZ², Metehan ÜNAL²,
Erkan BOSTANCI² and Mehmet Serdar GÜZEL²

¹Department of Artificial Intelligence and Data Engineering,
Ankara University, Ankara, TÜRKİYE

²Department of Computer Engineering, Ankara University, Ankara, TÜRKİYE

ABSTRACT. In the study, the implementation of 3D reconstruction of buildings using drones is explained. In this project, Airsim was used as the simulation environment and images were obtained from the simulation environment using OpenCV and the Meshroom software was run on these images and modeling was done in the computer environment. For real-world studies, the engineering faculty in Ankara University 50. Yıl Campus was modeled using photogrammetry technique. In the last part, the results of different modelling algorithms were compared.

1. INTRODUCTION

The 3D reconstruction of objects is a general scientific problem and fundamental technology of fields namely, Computer Aided Geometric Design (CAGD), computer graphics, computer animation, computer vision, medical imaging, computational science, virtual reality, digital media [1].

The 3D reconstruction is basically the process of capturing the appearance of objects and creating a realistic object from these data in the computer environment.

Yastikli et al. [2] mentioned the importance of 3D city model production, which is a very popular application that can be used in many different applications such as urban planning, facility management, indoor navigation, and 3D cadastre in recent years. Within the scope of the study, a 3D model was created from photogrammetric map and LIDAR data.

Keywords. Drones, 3D reconstruction, algorithms.

- ✉ kacici@ankara.edu.tr-Corresponding author;  0000-0002-3821-6419
- ✉ omert.erdal@gmail.com;  0000-0001-5256-3209
- ✉ alperen.ylmz2000@gmail.com;  0000-0002-6746-151X
- ✉ metehan.unal@ankara.edu.tr;  0000-0002-7545-2445
- ✉ ebostanci@ankara.edu.tr;  0000-0001-8547-7569
- ✉ mguzel@ankara.edu.tr;  0000-0002-3408-0083.

Today, the two most prominent reasons for using the internet are user-created content and social networks. A virtual environment that we encounter by hosting these two motifs is the 3D virtual worlds called metaverse, whose content is created by the users. In the last few years, owing to the creative content design possibilities these worlds offer, we have witnessed the launch of various metaverse experiments and the creation of an increasingly artistic and architectural design environment in these worlds [3]. We think that 3D restructuring techniques that will carry important structures, historical artifacts and cities around the world to digital environments will become much more popular in the future in this transformation and their use will become more common day by day.

In this study, with the help of the data obtained with the UAV LIDAR and UAV Photogrammetry techniques, which are the most common 3D reconstruction techniques, using a drone from the simulation environment, separate sensitive point clouds were created for these two methods, and then these point clouds were combined using the ICP algorithm and 3D point clouds were created. Then, for real-world studies, the engineering faculty in Ankara University 50th Year Campus was modeled using photogrammetry technique. The numerical data of the point clouds and models produced by using both methods and environments are given. In addition, using the available data set, the results of the 3D reconstructions made with the SIFT, AKAZE and DSPSIFT algorithms used in the photogrammetry technique were compared.

2. LITERATURE REVIEW

The Unmanned Aerial Vehicles (UAV) or drones became popular on recent years and the application area has become wider day by day [4,5]. In general drones can be used for transportation [6], military [7], education [8], tourism [9], and entertainment [10]. On the other hand, one of the most useful areas of the drones is imaging [11]. The capability of capturing wide range of areas makes drones very useful gadget for 3D reconstruction from images [12].

In the literature, there are 2 main approaches accepted in the field of 3D reconstruction. These are scanning and reconstruction with LIDAR systems using depth-based methods, and reconstruction using photogrammetry and computer vision methods [13]. In both methods, the purpose of the process is to collect a sufficiently large set of points in the computer environment by using the target and to recreate the target in the computer environment by making use of these collected points, but they have fundamental differences in terms of working principle.

Photogrammetry, by definition, is an information acquisition technology to assist the measurement and interpretation process of an object using high-resolution photographic images on a specific area or object.

For this study, it needs to be obtain a large number of photographs on an area, using a drone, to enable us to view the area from different vantage points. Similar to how the eyes use incoming information to provide depth perception in the human brain, photogrammetry uses these multiple viewpoints in images to create a 3D map. In this way, it allows to obtain various features by measuring each point on the map. It provides easier interpretation of the resulting 3D point cloud [14].

LIDAR (light detection and ranging) is technology for measuring the distance of an object or a surface using laser pulses. A LIDAR sensor sends pulses of laser light and measures the time it takes for these pulses to bounce off the surface and return. It works with the principle of obtaining the distance information of the object by using the reflected rays from these rays. It also measures the intensity of this reflection. Since it uses laser beams, it needs to work in constant motion. It constantly scans its surroundings and sends out many laser beams, creating a point cloud of millions of dots. In reconstruction methods using LIDAR, points are formed at certain distances according to the position of the scanner as data, and as a result of clustering these points, the target is reconstructed [15].

In many cases, a stand-alone LIDAR sensor will rival the cost of an entire photogrammetry data acquisition system. Precise geolocation lasers are more expensive than cameras. Therefore, it is crucial to evaluate your current and future applications to ensure that investing in LIDAR is the best decision [16].

The drone-based systems using photogrammetry are cost-effective. It gives you flexibility in where and when you get the data. Therefore, we can say that it is more useful.

Senol et al. [17], worked on the modeling of historical buildings using UAV photogrammetry and close-up photogrammetry methods. During their studies, they produced sensitive dense point clouds with UAV photogrammetry and close-up photogrammetry techniques of the Kanlıdivane basilica and modeled by combining the produced clouds. In their study, they stated that the shots taken from the ground caused difficulties in documenting the upper surfaces of the structures. In addition, they stated that the use of modern technologies in their studies is more cost and time efficient than traditional methods in documenting historical structures.

Ulvi et al. [18] used UAV photogrammetry techniques to model the Red Church in their work. In their study, they mentioned that in terrestrial photogrammetry, data could not be obtained at the desired density from some parts of the target, and they stated that UAV photogrammetry is suitable for reconstruction studies.

3. METHODOLOGY

In the past years, 3D reconstruction process performed using gadget on the ground. With the development in the area of UAV technology, the practicality of the reconstruction works was increased by loading the necessary equipment on these platforms and the problem of reaching the high points of the reconstructed objects were handled.

In the first step of the study, Airsim [19] simulation environment which is a medium to simulate different vehicles, gadget and drones, was utilized. It was developed by Microsoft as open source project and built on the Unreal Engine. The biggest advantage of the Airsim simulation environment is that the installation process is simple and it has the Python API, allowing rapid development using simulation. The properties of the tools used in Airsim can be determined by editing a json file. In this way, cameras and sensors can be easily attached to the vehicle. And the settings of these hardware can be made within the same file. Making basic adjustments easily has reduced the share of learning the use of the simulation environment in the time spent on the project, allowing more time to be devoted to the development of the 3D modeling, which is the main focus.

A virtual camera and LIDAR [20] system are installed on the aircraft in the simulation environment. In Figure 1, the parameters of the camera loaded on the UAV can be seen. Photographing process was carried out from a determined trajectory over the targeted area using a camera. Similarly, scanning from the orbit determined using LIDAR was performed. In Figure 2, the LIDAR parameters on the simulation are indicated.

```
"Cameras" : {  
  "high_res_bottom": {  
    "captureSettings" : [  
      {  
        "ImageType" : 0,  
        "Width" : 2048,  
        "Height" : 1080  
      }  
    ],  
    "X": 0, "Y": 0, "Z": 2,  
    "Roll": 0, "Pitch": -45, "Yaw" : 0  
  }  
},
```

FIGURE 1. Drone camera parameters in AirSim.

```
"LidarSensor2": {  
  "SensorType": 6,  
  "Enabled" : true,  
  "NumberOfChannels": 16,  
  "RotationsPerSecond": 10,  
  "PointsPerSecond": 90000,  
  "X": 0, "Y": 0, "Z": -1,  
  "Roll": 0, "Pitch": -45, "Yaw" : 0,  
  "VerticalFOVUpper": -45,  
  "VerticalFOVLower": 15,  
  "HorizontalFOVStart": -40,  
  "HorizontalFOVEnd": 40,  
  "DrawDebugPoints": true,  
  "DataFrame": "SensorLocalFrame"  
}
```

FIGURE 2. Drone LIDAR parameters in AirSim.

Since the first step of the study was carried out in the simulation environment, the control of the UAV was yielded using the Python libraries provided by the simulation environment. OpenCV [21] library was used to take photos from the camera. Simulation libraries were sufficient for recording LIDAR data. Structure From Motion technique was used in the photogrammetry part of the study, and Meshroom software, which is an open source and free software, was used to perform this technique. Scale-Invariant Feature Transform (SIFT) algorithm is used during point cloud generation. In this way, a 3D model was obtained using the images taken in the simulation environment through the Meshroom software.

Two separate point clouds were created from the LIDAR data and photographs obtained from the field, and these two point clouds were combined with the ICP algorithm. In this way, it was aimed to increase the accuracy of the model to be created from the data obtained, and at the same time, the data obtained from the two point clouds were compared. CloudCompare software was used for these processes.

After the simulation phase of the project was completed and the necessary findings were obtained, the 3D models obtained from various photogrammetry algorithms on the real world were compared. DJI Phantom 3 Professional Drone [22] given in Figure 3 was used for real world photogrammetry. In Table 1, the technical specifications of the drone used are given.



FIGURE 3. DJI Phantom 3 Professional Drone [22].

TABLE 1. DJI Phantom 3 Professional Drone Features.

Property	Value
Weight	1280 g
Flight time	23 min
GPS	Yes
autopilot	Yes
Maximum Flight Speed	16 m/s
Maximum Distance	6000 m
Camera Resolution	4000*3000
Camera Sensor	1/2.3" CMOS Effective pixels: 12.4 M
Lens	FOV 94° 20mm
Battery	4480mAh
Battery Type	LiPo 4S
Strength	68Wh



FIGURE 4. AirSimNH Map.



FIGURE 5. Panoramic view of buildings of Ankara University Faculty of Engineering.

For the simulation phase of the study, the AirsimNH map, which is included in the AirSim software by default, was used as the modeling area. This map is a neighborhood consisting of houses with gardens (Figure 4). There are no moving entities on the map.

After the simulation phase was completed, the Faculty of Engineering in Ankara University 50. Yıl Campus was selected for the real world phase (Figure 5).

The software to control the drone was developed in the simulation environment. In this way, it is ensured that the drone implements a trajectory of the determined diameter in a form directed towards the center of the orbit. Later, a camera and LIDAR were added to the drone, and configurations were made for these equipment to record at an inclined angle towards the orbital center. After the UAV equipment was adjusted, the flight was carried out over the designated house. During the flight, environmental photographs of the target area were taken and a LIDAR scan of the area was performed.

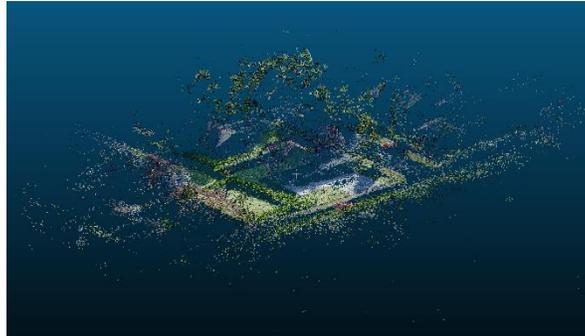


FIGURE 6. Colored point cloud formed by photogrammetry.

The captured photos were processed with the help of Meshroom software and a colored point cloud was obtained at the end of this process (Figure 6).

The color point cloud obtained and the point cloud obtained from the LIDAR scan were loaded into the CloudCompare software. The step to get the 3D reconstruction can be seen in Figure 7. At this stage, firstly, the cleaning process was applied on the point clouds. The state of the point cloud before and after the cleaning can be seen in Figure 8 and 9, respectively.

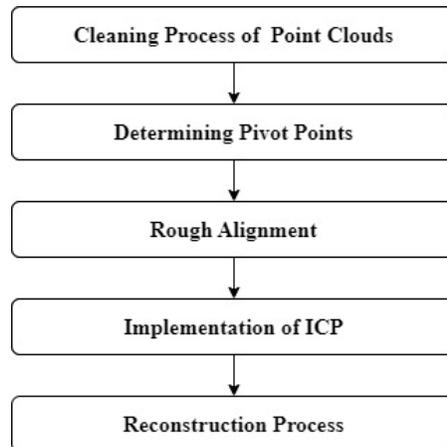


FIGURE 7. Cloud compare steps.

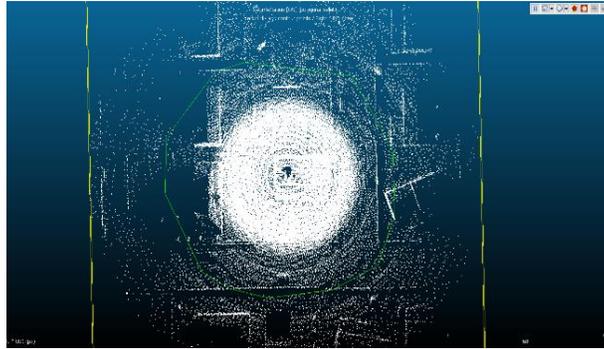


FIGURE 8. Point cloud before cleaning.

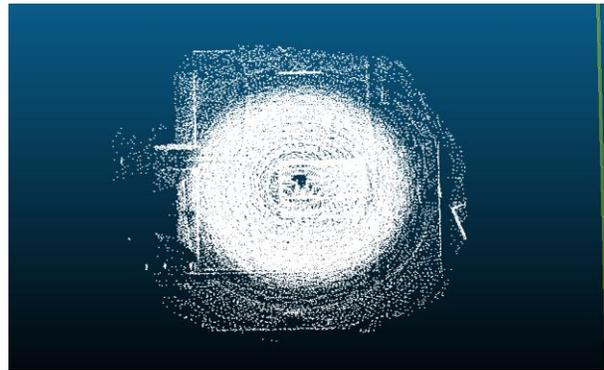


FIGURE 9. State of the point cloud after cleaning.

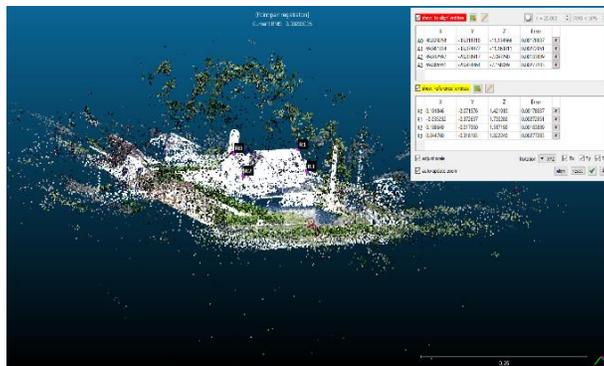


FIGURE 10. Rough alignment of LIDAR and photogrammetry point clouds.

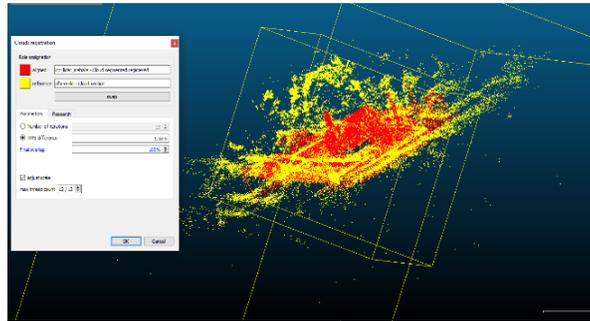


FIGURE 11. Aligning LIDAR and photogrammetry point clouds using ICP algorithm.

The cleared point clouds were first subjected to a rough alignment and proportioning process by manually determining the pivot points (Figure 10).

By applying the ICP process on the roughly aligned point clouds, precise alignment of the point clouds with each other was achieved (Figure 11).

The point cloud obtained by the photogrammetry technique consists of X points. The point cloud obtained by the LIDAR scanning technique consists of X points. The point cloud obtained as a result of combining two point clouds using the ICP algorithm consists of X points.

At this stage, the interpolation process for the colors was carried out by taking the photogrammetry point cloud as a guide for the coloration of the LIDAR point cloud.

The obtained colored LIDAR point cloud was combined with the photogrammetry point cloud and at the end of this process, a single colored point cloud was obtained (Figure 12).



FIGURE 12. Concatenated point cloud using ICP algorithm.



FIGURE 13. A photograph from the data set used for modelling the faculty building.



FIGURE 14. Reconstructed model created by SIFT algorithm.

After the simulation phase was completed, real-world studies were started. At this stage, first of all, photography was taken on the campus using a drone. (Figure 13).

The photographs taken were processed using the SIFT algorithm in the meshroom software, and as a result, a 3D model of the faculty building was created. Below is a sample photo of the 3D model created with the SIFT algorithm. (Figure 14).

4. EXPERIMENTAL CONFIGURATION AND RESULTS

It has been concluded that the use of aircraft provides sufficient viewing angles to these points and is appropriate to use, since there are difficulties in taking pictures of high points and thus creating a 3D model of the structure during the studies.

Today, with the development of technology, technologies targeting Metaverse and similar virtual worlds are becoming popular day by day. Investments are made in technologies that will carry important structures, historical artifacts and cities around the world to digital environments. It has been concluded that the use of 3D

reconstruction techniques for this transformation is appropriate and facilitates these processes.

A data set of 66 photographs was used to model the faculty building. SIFT, DSPSIFT and AKAZE algorithms were tested on the data set and point clouds consisting of 125,498 points for SIFT, 147,554 points for DSPSIFT and 111,589 points for AKAZE were obtained. In the case of converting these clouds into models, models consisting of 2,557,802 polygons for SIFT, 2,704,615 polygons for DSPSIFT and 2,577,180 polygons for AKAZE were produced. In the scenario where three algorithms are used together, a point cloud consisting of 366,162 points and a model consisting of 2,522,214 polygons were obtained. The results can be seen in Table 2.

TABLE 2. Comparison of algorithms used in photogrammetry technique.

Algorithms	Number of Points	Number of Polygons
SIFT	125,498	2,557,802
AKAZE	111,589	2,577,180
DSPSIFT	147,554	2,704,615
SIFT+AKAZE +DSPSIFT	366,162	2,522,214

Author Contribution Statements Authors are equally contributed to the paper. All authors read and approved the final copy of the manuscript.

Declaration of Competing Interests The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement This research did not receive any specific grant from funding agencies in public or not-for-profit sectors.

REFERENCES

- [1] Moons, T., Van Gool, L., Vergauwen, M., 3D reconstruction from multiple images part 1: Principles, *Found. Trends Comput. Graph. Vis.*, 4 (4) (2010), 287-404.
- [2] Yastıklı, N., Çetin, Z., Üçok, U., Koçdemir, K. H., Fotogrametrik harita ve LiDAR

- verileri ile 3B kent modeli üretimi, *TMMOB Harita ve Kadastro Mühendisleri Odası, 16. Türkiye Harita Bilimsel ve Teknik Kurultayı*, Ankara, (2017).
- [3] Tasa, U. B., İçeriği Kullanıcılar Tarafından Oluşturulan 3 Boyutlu Sanal Dünyalarda Sanat ve Mimari Tasarım: Second Life® Üzerine Bir Vaka Çalışması, (2009). Master Thesis, Yıldız Teknik Üniversitesi, Türkiye.
- [4] Abualigah, L., Diabat, A., Sumari, P., Gandomi, A. H., Applications, deployments, and integration of internet of drones (IoD): A review, *IEEE Sens. J.*, 21 (22) (2021), 25532-25546.
- [5] Ayamga, M., Akaba, S., Nyaaba, A. A., Multifaceted applicability of drones: A review, *Technol. Forecast. Soc. Change*, 167 (2021), 120677, <https://doi.org/10.1016/j.techfore.2021.120677>.
- [6] Lamptey, E., Serwaa, D., The use of zipline drones technology for COVID-19 samples transportation in Ghana, *HighTech Innov. J.*, 1 (2) (2020), 67-71, <https://doi.org/10.28991/HIJ-2020-01-02-03>.
- [7] Konert, A., Balcerzak, T., Military autonomous drones (UAVs) - from fantasy to reality. Legal and Ethical implications, *Transp. Res. Proc.*, 59 (2021), 292-299, <https://doi.org/10.1016/j.trpro.2021.11.121>.
- [8] Bai, O., Chu, H., Drones in education: A critical review, *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, 12 (11) (2021), 1722-1727.
- [9] Unal, M., Bostanci, E., Sertalp, E., Distant augmented reality: Bringing a new dimension to user experience using drones, *Digit. Appl. Archaeol. Cult. Heritage*, 17 (2020), e00140, <https://doi.org/10.1016/j.daach.2020.e00140>.
- [10] Unal, M., Unal, F. Z., Bostanci, E., Guzel, M. S., Augmented reality and new opportunities for cultural heritage, *Augmented Reality in Tourism, Museums and Heritage*, Springer, Cham, 2021, 213-225.
- [11] Andriolo, U., Gonçalves, G., Rangel-Buitrago, N., Paterni, M., Bessa, F., Gonçalves, L. M. S., et al., Drones for litter mapping: An inter-operator concordance test in marking beached items on aerial images, *Mar. Pollut. Bull.*, 169 (2021), 112542, <https://doi.org/10.1016/j.marpolbul.2021.112542>.
- [12] Aydın, M., Bostancı, E., Güzel, M. S., Kanwal, N., Multiagent systems for 3D reconstruction applications, *Multi Agent Systems - Strategies and Applications*, IntechOpen, London, 2020, <https://doi.org/10.5772/intechopen.88460>.
- [13] Lee, J., Hong, S., Cho, H., Park, I., Cho, H., Sohn, H.-G., Accuracy comparison between image-based 3D reconstruction technique and terrestrial LIDAR for as-built BIM of outdoor structures, *J. Korean Soc. Surv. Geod. Photogramm. Cartogr.*, 33 (2015), 557-567, <https://doi.org/10.7848/ksgpc.2015.33.6.557>.
- [14] Wingtra, Drone photogrammetry vs. LIDAR: what sensor to choose for a given application, (2021). Available at: <https://wingtra.com/drone-photogrammetry-vs-LIDAR/>. [Accessed October 2022].

- [15] Milioto, A., Vizzo, I., Behley, J., Stachniss, C., Rangenet++: Fast and accurate LiDAR semantic segmentation, *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (2019), 4213-4220.
- [16] Raj, T., Hashim, F. H., Huddin, A. B., Ibrahim, M. F., Hussain, A., A survey on LiDAR scanning mechanisms, *Electronics*, 9 (5) (2020), 741.
- [17] Senol, H. I., Yiğit, A. Y., Kaya, Y., Ulvi, A., İHA ve yersel fotogrametrik veri füzyonu ile kültürel mirasın 3 boyutlu (3B) modelleme uygulaması: Kanlıdivane örneği, *Türkiye Fotogrametri Dergisi*, 3 (1) (2021), 29-36.
- [18] Ulvi, A., Yakar, M., Yiğit, A., Kaya, Y., İHA ve yersel fotogrametrik teknikler kullanarak aksaray kızıl kilisenin 3b modelinin ve nokta bulutunun elde edilmesi, *Geomatik*, 5 (1) (2020), 19-26, <https://doi.org/10.29128/geomatik.560179>.
- [19] Microsoft Research Github Web Page, "Airsim," (2022). Available at: <https://microsoft.github.io/AirSim/>. [Accessed October 2022].
- [20] Reutebuch, S. E., Andersen, H. E., McGaughey, R. J., Light detection and ranging (LIDAR): An emerging tool for multiple resource inventory, *J. For.*, 103 (6) (2005), 286-292.
- [21] Bradski, G., The OpenCV Library, *Dr. Dobb's Journal of Software Tools*, (2000).
- [22] DJI, DJI Drones, (2022). Available: <https://www.dji.com/>. [Accessed October 2022].

IONIZATION AND PHONON PRODUCTION BY ^{10}B IONS IN RADIOTHERAPY APPLICATIONS

Fatih EKINCI¹

¹Institute of Nuclear Sciences, Ankara University, Ankara, TÜRKİYE

ABSTRACT. The therapeutic use of heavy ions has received much attention due to their physical and radiobiological properties. Thanks to these features of heavy ion radiotherapy, radiation in tissues close to critical tissues can reduce LET while allowing an increase in LET in tumors. Selection of biomaterials closest to the tissue is critical to measure the accuracy of this LET transfer. The accuracy of LET and radiological features measured in phantoms created from biomaterials selected according to the characteristics of the target tissue is very important for human life. For this reason, the research of polymeric materials, which is the closest biomaterial to soft tissue and therefore phantom material, has increased recently. In this study, ionization to the polymeric biomaterials closest to the soft tissue in boron therapy application, and phonon release from all interactions were investigated and analyzed. This analysis was performed using MC-based TRIM simulation. In the analysis, the Bragg peak closest to the soft tissue was 7.2% and PMMA was the phonon release from all interactions. It has been observed that the phonon production in phantoms results from ions on average 30% and recoils interactions 70%. The main novelty that this study will provide to the literature is to consider the phonon interactions as well as the ionization interactions. Thus, apart from proton and carbon, the most ideal polymeric biomaterial to be used instead of soft tissue was evaluated by calculating all interactions. Thus, it is aimed to determine the most ideal phantom material.

1. INTRODUCTION

Heavy ion therapy has been the focus of radiation oncology for over 60 years because of its superior physical and biological properties [1, 2]. In addition to the proton [4]

Keywords. B ion therapy, polymeric biomaterials, Bragg cure, phonon, TRIM Monte Carlo.

✉ fatih.ekinci@gsb.gov.tr-Corresponding author  0000-0002-1011-1105.

and carbon therapy centers that are widely used around the world, researchers have recently focused on the use of helium and oxygen ion beams in therapy [5, 6]. Before clinical applications of heavy ion therapy, accurate dose calculations and dosimetric verification procedures, including heavy ion beam delivery, are important. Therefore, most of the clinical knowledge in heavy ion therapy has been formed by radiotherapy planning and dose validations based on semi-analytical algorithms [7]. The selection and beam energies of each new heavy ion species mainly depend on the ionization interactions of the ions in question [8]. Thus, the ionization properties of heavy ion beams such as ^1H , ^4He , $^{6,7}\text{Li}$, ^8Be , ^{10}B , ^{12}C , ^{14}N and ^{16}O on the target at therapeutic energy levels were investigated on the phantom with the help of different methods [9, 10].

It is accepted that the accuracy of the phantom close to clinical results is related to the fact that the building materials that make up the phantom are equivalent to the target tissue [11]. The International Atomic Energy Agency (IAEA) recommended water as the phantom building material for radiation measurements that are closest to the standard soft tissue, easily available and repeated [12]. In addition to these properties, the most important reason why water is a widely used biomaterial is that its bulk density is the closest material to soft tissue [13]. In addition to general properties, stopping powers, linear energy transfer (LET), Bragg curve properties and nuclear interaction results are important in the evaluation of biomaterials that make up phantoms radiologically [11, 14]. However, although less quantified and understood, recoil and phonon production are important in the selection of biomaterials in interactions [14, 15]. When all these properties are evaluated, one of the biomaterials close to soft tissue, such as water, is polymeric [15].

In this study, ionization, recoils, phonon and lateral scattering values of helium, lithium, beryllium and boron (^{10}B) ions formed by water phantom were calculated with the help of Monte Carlo (MC) Transport of Ions in Matter (TRIM) simulation program. In these calculations, the data obtained for each selected heavy ion were compared with each other. First of all, in these comparisons, it was tried to determine the ion that creates the most LET energy. Then the ion generating the lowest recoils and photon energy was determined. Finally, it is aimed to determine from which heavy ion the least lateral scattering value comes from.

2. SECTION

In the absence of an experimental ^{10}B beamline, MC-based simulations were used to generate reference values for our study. In this direction, it is important to investigate biological experiments with the help of simulation and to make experimental validation of the data obtained. One of these simulations, TRIM, is an MC-based system widely used in ion beam implantation and processing calculations [16].

TRIM can calculate radiological interaction on complex targets with layered structure. TRIM provides different options for calculating radiological interactions on a target based on the type of data outputs required. The full damage rank (F-C) mod can track every knockback atom until its energy drops below the displacement threshold energy (E_d) of any target atom. Thus, all collisions of the ion beam towards the target can be calculated and analysed. The fast damage calculation (K-P) mode can only follow the path of the ion beam, can be used when little attention is required to details of damage to the target or surface spraying. Damage calculated with this option is a rapid statistical method based on the Kinchin-Pease formalism "K-P" [17]. The K-P theory used by the TRIM system was first proposed by Kinchin and Pease, expanded by Lindhard, and later used by Norgett, Robinson and Torrens (NRT) [18].

"Detailed Calculation with Full Damage Cascades" type was selected in the calculations from the display window of the TRIM program. The particle number of the heavy ion beam was entered as 10^5 particles with the "Total Number of Ions" tab. Calculation outputs from the "Output disk files" tab; Ion range, recoils, sputtered atoms and collision details output files were selected. In order to compare the selected heavy ions with each other, four different ion beam energies were determined as 80 MeV/u, 100 MeV/u, 120 MeV/u and 140 MeV/u. The phantom type was created from the "Compound Dictionary" tab as water and its geometry from the "Add New Layer" section. Of the polymeric materials used as biomaterials in this study; Phantoms formed from two polymeric biomaterials such as polymethylmetacrylate (PMMA) and polystyrene (PS), water and soft tissue (ST) materials were used. The phantoms created from these biomaterials were bombarded with a ^{10}B ion pencil beam with 10^5 particles at therapeutic energies (80, 100, 120 and 140 MeV/u) by forming a single-layer sheet phantom with a width of 15 cm and a length of 15 cm.

3. RESULTS

The Bragg peak position formed by the ^{10}B heavy ion with 80-140 MeV/u energy obtained from this study in selected phantoms is shown in Figure 1. As the energy of the ^{10}B ion beam increases, the Bragg peak position is shifted in all phantoms, as expected. The longest Bragg peak range value of the ^{10}B ion beam was seen in the PMMA polymeric phantom. The next longest bracket peak range occurred in the water phantom. For each 20 MeV/u energy increase of the ^{10}B ion beam, the average Bragg peak range increases of 1.5 cm in the ST phantom, 1.4 cm in the PMMA phantom, 1.35 cm in the water phantom and 1.3 cm in the PS phantom, respectively. In terms of Bragg peak range, the closest phantom to soft tissue was PMMA with

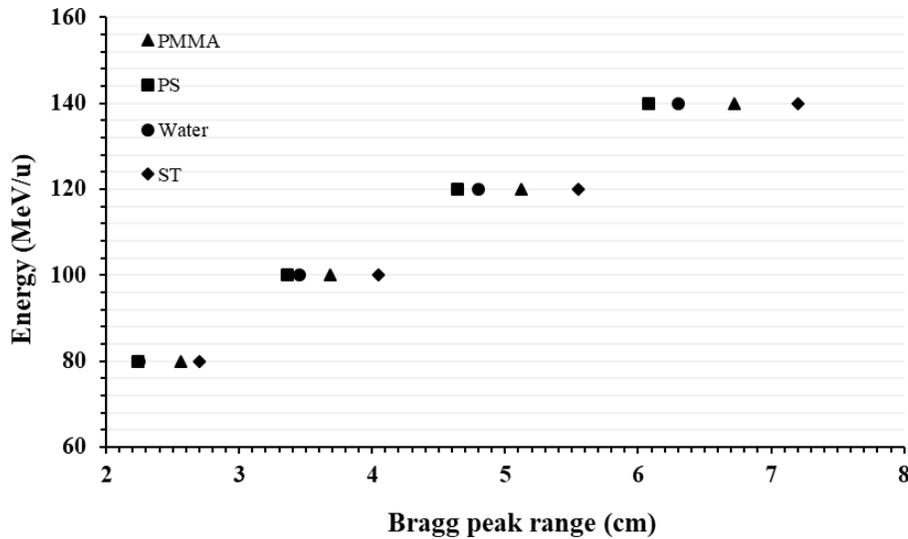


FIGURE 1. Bragg peak range (cm) created by the ^{10}B ion beam in four phantoms.

7.2% difference. In other phantoms, this difference is respectively; 14.4% in water and 16.5% in PS polymeric material.

Phonons are formed during and at the end of the interaction of ionization and recoils caused by the ^{10}B ion beam in plate phantoms. All phonon formations consisting of interactions received in the TRIM program are analyzed and given in Table 1. In order for the given phonon formations to be meaningful, they are presented by multiplying by 10^3 . Looking at the presented data, it was seen that the biggest contribution to phonon production came from the recoils interactions. While PMMA was the closest biomaterial to ST in phonon contribution from ions, PS was the closest biomaterial to ST in phonon contribution from recoils interactions. It was observed that as the energy increased, the phonon contribution from the recoils interaction generally increased. In general, it was observed that the biomaterial that provides the closest phonon production to ST is PMMA.

In this study, Bragg peak range of ^{10}B ion beam with therapeutic energy and phonon values consisting of all interactions were calculated in 4 different phantoms. In the selected phantom materials, it was tried to find the closest biomaterial to the soft tissue in ^{10}B ion treatment. Thus, the interactions of the ^{10}B ion beam in soft tissue, water and polymeric biomaterials were considered. The biological activity from these interactions may differ between the selected ion beam types [19]. These differences are not only limited to the type of ion, but each type of treatment can be

customized depending on whether the target is hard or soft tissue [20]. At this point, it is considered to be an important issue to investigate selected soft tissue phantom biomaterials. Thus, ST water and two different polymeric biomaterials were selected for the phantom material and compared with ST. By using heavier ion beams apart from the 10B ion beam presented in this study, it provides the presence of an appropriate level of dose that ensures tumour control while sparing healthy tissues at risk [21]. It is noteworthy that ^4He , $^6,7\text{Li}$, ^8Be , ^{10}B , ^{12}C , ^{14}N or ^{16}O beams are preferred over He beams, which increases the likelihood of treatment in clinical practice, as suggested in similar studies [22, 23]. It has been observed that as the atomic weight of the ions forming the beam increases, they exhibit advantageous physical properties [24-26]. This provides possible clinical advantages for radiation-resistant tumours [23]. In this study, the contribution of heavy ions recoil interaction [1, 15, 27] to total phonon production was investigated.

TABLE 1. Phonon values ($\text{eV}/\text{A} \times 10^3$) formed by ion and recoil interactions formed by ^{10}B ion beams in four phantoms and their percentage (%) contribution to this value.

Phantom	80 MeV/u				100 MeV/u				120 MeV/u				140 MeV/u			
	ion		recoil		ion		recoil		ion		recoil		ion		recoil	
	eV/A	%	eV/A	%	eV/A	%	eV/A	%	eV/A	%	eV/A	%	eV/A	%	eV/A	%
PMMA	0.804	31	2.629	69	0.555	32	1.057	68	0.655	26	1.651	74	0.594	27	1.887	73
PS	0.859	47	2.678	53	0.793	31	2.553	69	0.774	31	2.511	69	0.771	13	2.506	87
Water	0.801	26	3.133	74	1.038	31	3.333	69	1.022	26	3.567	74	0.987	34	3.483	64
ST	0.901	27	3.346	73	0.403	13	3.062	87	0.836	26	3.214	74	0.632	34	1.881	66

4. CONCLUSIONS

In this study, radiological interactions of ^{10}B ion beam in ST, water and polymeric biomaterial were investigated with the help of MC TRIM. From these interactions, ionization and the phonon state formed as a result of all interactions have been revealed. As a result of the obtained results and studies in the literature, biomaterials that are considered close to soft tissue were compared. It is recommended that these calculations be made for different heavy ions as well. Considering the importance of investigating biomaterials close to tissues in radiotherapy, it has been evaluated to be made in different biomaterial types.

Declaration of Competing Interests The authors declare no conflict of interest.

REFERENCES

- [1] Ekinci, F., Bostanci, E., Güzel, M. S., Dagli, O., Effect of different embolization materials on proton beam stereotactic radiosurgery Arteriovenous Malformation dose distributions using the Monte Carlo simulation code, *J. Radiat. Res. Appl. Sci.*, 15 (3) (2022), 191-197, <https://doi.org/10.1016/j.jrras.2022.05.011>.
- [2] Senirkentli, G. B., Ekinci, F., Bostanci, E., Güzel, M. S., Dagli, Ö., Karim, A. M., Mishra, A., Therapy for mandibula plate phantom, *Healthcare*, 9 (2021) 167, <https://doi.org/10.3390/healthcare9020167>.
- [3] Durante, M., Loeffler, J. S., Charged particles in radiation oncology, *Nat. Rev. Clin. Oncol.*, 7 (2010), 37-43, <https://doi.org/10.1038/nrclinonc.2009.183>.
- [4] PTCOG, Particle therapy co-operative group, (2020). Available at: <https://www.ptcog.ch>. [Accessed 20 October 2020].
- [5] Tessonnier, T., Böhlen, T. T., Ceruti, F., Ferrari, A., Sala, P., Brons, S., Haberer, T., Debus, J., Parodi, K., Mairani, A., Dosimetric verification in water of a Monte Carlo treatment planning tool for proton, helium, carbon and oxygen ion beams at the Heidelberg Ion Beam Therapy Center, *Phys. Med. Biol.*, 62 (2017), 6579-6594, <https://doi.org/10.1088/1361-6560/aa7be4>.
- [6] Tessonnier, T., Mairani, A., Brons, S., Haberer, T., Debus, J., Parodi, K., Experimental dosimetric comparison of ^1H , ^4He , ^{12}C and ^{16}O scanned ion beams, *Phys. Med. Biol.*, 62 (2017), 3958-3982, <https://doi.org/10.1088/1361-6560/aa6516>.
- [7] Kozłowska, W. S., Böhlen, T. T., Cuccagna, C., Ferrari, A., Fracchiolla, F., Georg, D., Magro, G., Mairani, A., Schwarz, M., Vlachoudis, V., FLUKA particle therapy tool for Monte Carlo independent calculation of scanned proton and carbon ion beam therapy, *Phys. Med. Biol.*, 64 (2019), 075012, <https://doi.org/10.1088/1361-6560/ab02cb>.
- [8] Lundkvist, J., Ekman, M., Ericsson, S., Jonsson, B., Glimelius, B., Proton therapy of cancer: Potential clinical advantages and cost-effectiveness, *Acta. Oncol.*, 44 (2005), 850-861, <https://doi.org/10.1080/02841860500341157>.
- [9] Kempe, J., Gudowska, I., Brahme, A., Depth absorbed dose and LET distributions of therapeutic H^1 , He^4 , Li^7 and C^{12} beams, *Med. Phys.*, 34 (2007), 183-192, <https://doi.org/10.1118/1.2400621>.
- [10] Kantemiris, I., Karaiskos, P., Papagiannis, P., Angelopoulos, A., Dose and dose averaged LET comparison of H^1 , He^4 , Li^6 , Be^8 , B^{10} , C^{12} , N^{14} and O^{16} ion beams forming a spread-out Bragg peak, *Med. Phys.*, 38 (2011), 6585-6591, <https://doi.org/10.1118/1.3662911>.
- [11] Lourenço, A., Wellock, N., Thomas, R., Homer, M., Bouchard, H., Kanai, T., MacDougall, N., Royle, G., Palmans, H., Theoretical and experimental characterization of novel water-equivalent plastics in clinical high-energy carbon-ion beams, *Phys. Med. Biol.*, 61 (21) (2016), 7623-7638, <https://doi.org/10.1088/0031-9155/61/21/7623>.

- [12] Arib, M., Medjadj, T., Boudouma, Y., Study of the influence of phantom material and size on the calibration of ionization chambers in terms of absorbed dose to water, *J. Appl. Clin. Med. Phys.*, 7 (2006), 55-64, <https://doi.org/10.1120/jacmp.v7i3.2264>.
- [13] Samson, D. O., Jafri, M. Z. M., Shukri, A., Hashim, R., Sulaiman, O., Aziz, M. Z. A., Yusof, M. F. M., Measurement of radiation attenuation parameters of modified defatted soy flour–soy protein isolate-based mangrove wood particleboards to be used for CT phantom production, *Radiat. Environ. Biophys.*, 59 (2020), 483–501, <https://doi.org/10.1007/s00411-020-00844-z>.
- [14] Ekinci, F., Bölükdemir, M. H., The effect of the second peak formed in biomaterials used in a slab head phantom on the proton Bragg peak, *J. Polytech.*, 23 (1) (2019), 129-136, <https://doi.org/10.2339/politeknik.523001>.
- [15] Ekinci, F., Bostanci, E., Dagli, O., Guzel, M. S., Analysis of Bragg curve parameters and lateral straggle for proton and carbon beam, *Commun. Fac. Sci. Univ. Ank. Series A2-A3*, 63 (1) (2021), 32-41, <https://doi.org/10.33769/aupse.864475>.
- [16] Stoller, R., Toloczko, M., Was, G., Certain, A., Dwaraknath, S., Garner, F., On the use of SRIM for computing radiation damage exposure, *Nucl. Instrum. Methods Phys. Res. B*, 310 (2013), 75-80, <https://doi.org/10.1016/j.nimb.2013.05.008>.
- [17] Kinchin, G. H., Pease, R. S., The displacement of atoms in solids by radiation, *Rep. Prog. Phys.*, 18 (1955), 1-51, <https://doi.org/10.1088/0034-4885/18/1/301>.
- [18] Ziegler, J. F., Biersack, J. P., Ziegler, M. D., SRIM-The Stopping and Range of Ions in Matter, SRIM Co., Boston, 2008.
- [19] Loeffler, J. S., Durante M., Charged particle therapy optimization. challenges and future directions, *Nat. Rev. Clin. Oncol.*, 10 (2013), 411-424, <https://doi.org/10.1038/nrclinonc.2013.79>.
- [20] Uhl, M., Mattke, M., Welzel, T., Roeder, F., Oelmann, J., Habl, G., Jensen, A., Ellerbrock, M., Jakel, O., Haberer, T., Herfarth, K., Debus, J., Highly effective treatment of skull base chordoma with carbon ion irradiation using a raster scan technique in 155 patients: first long-term results, *Cancer*, 120 (2014), 3410-3417, <https://doi.org/10.1002/cncr.28877>.
- [21] Schulz-Ertner, D., Tsujii, H., Particle radiation therapy using proton and heavier ion beams, *J. Clin. Oncol.*, 25 (2007), 953-964, <https://doi.org/10.1200/JCO.2006.09.7816>.
- [22] Grun, R., Friedrich, T., Kramer, M., Zink, K., Durante, M., Engenhart-Cabillic, R., Scholz, M., Assessment of potential advantages of relevant ions for particle therapy: a model based study, *Med. Phys.*, 42 (2015), 1037-1047, <https://doi.org/10.1118/1.4905374>.
- [23] Phillips, T. L., Fu, K. K., Curtis, S. B., Tumor biology of helium and heavy ions, *Int. J. Radiat. Oncol. Biol. Phys.*, 3 (1977), 109-113, [https://doi.org/10.1016/0360-3016\(77\)90236-x](https://doi.org/10.1016/0360-3016(77)90236-x).

- [24] Strobele, J., Schreiner, T., Fuchs, H., Georg, D., Comparison of basic features of proton and helium ion pencil beams in water using GATE, *Z Med. Phys.*, 22 (2012), 170-178, <https://doi.org/10.1016/j.zemedi.2011.12.001>.
- [25] Orecchia, R., Krengli, M., Jerezek-Fossa, B. A., Franzetti, S., Gerard, J. P., Clinical and research validity of hadrontherapy with ion beams, *Crit. Rev. Oncol. Hematol.*, 51 (2004), 81-90, <https://doi.org/10.1016/j.critrevonc.2004.04.005>.
- [26] Karger, C. P., Jakel, O., Palmans, H., Kanai, T., Dosimetry for ion beam radiotherapy, *Phys. Med. Biol.*, 55 (2010), R193-R234, <https://doi.org/10.1088/0031-9155/55/21/R01>.
- [27] Qi, M., Yang, Q., Chen, X., Duan, J., Yang, L., Fast calculation of Mont Carlo ion transport code, *J. Phys. Conf. Ser.*, 1739 (2021), 012030, <https://doi.org/10.1088/1742-6596/1739/1/012030>.

ONTOLOGY DEVELOPMENT FOR WEB SERVICES TO BE USED WITHIN THE SCOPE OF REMOTE MONITORING PROJECT

Kader GÜRCÜOĞLU¹, Tunç MEDENİ², Tolga MEDENİ², Mehmet Serdar GÜZEL³,
Halil ARSLAN¹

¹ e-Government and Public Transformation Program, Ankara Yıldırım Beyazıt University,
Ankara, TÜRKİYE

² Management Information System, Ankara Yıldırım Beyazıt University,
Ankara, TÜRKİYE

³ Computer Engineering, Ankara University,
Ankara, TÜRKİYE

ABSTRACT. In today's society, as the digital transformation has become widespread rapidly; information technologies also started to develop themselves quickly along with this prevalence. This rapid development and transformation bring about new and different requirements. Situations like reuse of the information, the ability to integrate the obtained information and sharing it, among others, have pushed Semantic Web to the forefront, especially scientifically. Semantic Web, which provides the communication of a machine with other machines, gets a great attention especially in today's digital age. Probably because of this, significant works have been done on ontology development method and ontology based systems started to be advanced over the last decade. Ontology development methods or ontology based systems play a key role in the integrity, being shared and management of the data. Ontology development method, which is of vital importance in reusability and expandability of real time monitoring systems, is in a position to be accommodable to many architectural systems like Deep Learning architecture at the same time.

1. INTRODUCTION

Ontology development studies, which have a serious importance in semantic web-

Keywords. Deep learning, semantic web, ontology development method, image processing module.

✉ kadrgrcoglu33@gmail.com-Corresponding author;  0000-0002-4527-1993

✉ tuncmedeni@ybu.edu.tr;  0000-0002-2964-3320

✉ tolgamedeni@ybu.edu.tr;  0000-0002-0642-7908

✉ mguzel@ankara.edu.tr;  0000-0002-3408-0083

✉ halilarslan5006@gmail.com;  0000-0003-3959-0872.

based systems, ensure that the applications used to intersect with different fields. Ontology development study aims to manage data in a wise way. In addition to that, Ontology development study is a solution tool in data integration studies. Since applications contain different systems within itself, it is very difficult to contact with different applications. At this point, ontology development studies are needed. Ontology development emerges in accordance with certain requirements and is designed in accordance with the next new requirements. It is a system that starts to be used frequently especially in the data subtitles due to this feature.

The main point of the ontology development method, which provides the key task in data by providing data integrity, consists of a reflection of software engineering and is also used in almost every field. The basic function of the semantic Web actually allows the communication of machines with other machines by using formal semantic science to share and reuse information [1]. Therefore, another purpose of ontology development work is to provide inter-machine communication. In an ontology infrastructure design, the main tasks are distributed among the functions and the information from the user interface based on the distribution of tasks is transferred to the servers by the ontology regulator. Optionally, ontology can make all operations on the server. And so that ontology can only act like an interface that conveys the resulting results to the user [2].

The rapid development of information technologies has revealed the new infrastructure requirements of semantic web-based infrastructure. One of the areas that needs this infrastructure requirement is image processing modules in remote monitoring systems. In these modules, deep learning architecture is of great importance. On the other hand, ontology-based system studies prepared for use in deep learning architecture have just begun to become widespread.

In this study, an ontology-based image monitoring system design will be introduced. In the first part, information about the image processing module and deep learning will be given. In the second part, the concept of ontology development and ontology definition languages will be mentioned. In the third part, under the title of ontology development methodology, the necessary details will be conveyed and finally a general evaluation will be made in the conclusion section. In the third part, the necessary details about how the process should work under the title of ontology development methodology will be given and finally a general evaluation will be made in the conclusion part.

2. IMAGE PROCESSING MODULE AND DEEP LEARNING

In the modern era, there were practices for monitoring industrial parameters such as temperature reading, hardware supply, pressure level, inventory monitoring and management of the relevant machines. The change in the data collected with these

applications can lead to serious consequences. Not only these parameters are important, but also for monitoring the position of placed objects. The monitoring of these parameters has changed every passing day since the last few years. In here, various researches are conducted on the use of software technologies such as image processing and machine learning to monitor some industrial parameters.

Manual power is needed to monitor parameters of voltage, different gases, etc. in industry. If these parameters are not correctly monitored, abnormal conditions may occur as a result. This is one of the important issues in the industrial sector. Using manual force to control these abnormal conditions in harmful situations is quite risky. Therefore, manual power alone cannot always be relied upon. There is a need for methods that can automate processes. In recent years, monitoring systems have been far from solutions based on manual power. Instead, the system can be automated using up to date technologies such as image processing that allows monitoring on a per second basis during the day. Thanks to the images uploaded to the system, this data can be seen on the computer system and can be viewed from any remote locations that can be accessed to the website. The biggest advantage of the development of such a system is that it minimizes the risks of abnormal states.

Even though the first studies on deep learning are based on the past, one of the main reasons for successful use in recent years is that there is enough data. The algorithms that train deep learning models used in complex tasks are almost the same as the learning algorithms used to solve toy problems in the 1980s. However, the models we have prepared with these algorithms have made changes that simplify the education of very deep architecture. In addition, another important new development is to provide the resources they need to succeed in these algorithms today. The first data that constitutes the first of these sources is achieved by increasing the digitalization of society. With the increase in the activities carried out on computers, the transactions are more recorded. Since computers are connected to more networks, it has become easier to centralize these records and make them a suitable data set for machine learning applications. This increasing data structure has created a new field called "Big Data" in recent years. With big data, machine learning has become much easier. Another reason why deep learning is popular is because computational resources are available today to run larger models. With the introduction of hidden layers in artificial neural networks, the processor capacity has increased for the memory and computational memory used. The network, which is deepened by increasing the number of hidden layers, creates a faster computer need with larger memory. With the development of big data and GPUs, it has been possible to design different Deep Learning models. These designed models make the learning process itself without user-specified features from the input data. This learning process is obtained by discovering different features of data in different layers. The basic model of these architectures is considered Convolutional Neural Networks (CNN). CNNs

are successfully applied in image classification, object recognition, image segmentation, etc. problems. CNNs are the improved state of artificial neural networks. The network that deepens as a result of further increasing the number of hidden layers in artificial neural networks can be called CNN. This depth in CNN was performed by using 2-dimensional filters. In addition to this difference in depth, CNNs perform learning in a hierarchical structure. Finally, the main difference that distinguishes CNNs from my artificial neural networks is the Dropout method used by CNNs to prevent memorization during the training of the deepening network structure. This method aims to prevent memorization randomly by removing some nodes of the network in each iteration during the training phase. Today, deep learning method is used in most of the image processing applications. Deep Learning, which provides very successful results in both industrial and academic studies, will also be implemented in this project. As a basic approach, object detection will be tried to be made and CNN will be used for the solution of this problem.

Objectnet (<https://objectnet.dev/>) data set will be used for the training and verification of the network to be used. Objectnet is a real large data test set for object recognition by control that object backgrounds, rotations and imaging perspectives are random. Similar to the Imagenet, there are 50,000 test data in the 313 object class. One of the important challenges encountered in automatic classification is the emergence of a new class after the education stage. In recent years, researchers have proposed ZSL approaches to overcome this problem. ZSL is a simple but effective algorithm that proposes linear code solution in these approaches. The performance of the relevant algorithm is basically dependent on the selection of parameters. Meta intuitive algorithms are considered as approaches that allow to approach optimization problems with high success. In this context, a problem -specific model and parameter training approaches will be recommended. Developments will be performed with Python language. The problem of estimation of human movements is an area studied in the literature. However, there is a need for a new model unique to the related problem. In this context, the aim is to develop a CNN-based model. If the adaptation of pretrained models to the relevant problem does not produce the desired results, a hybrid method is intended to be used to combine the respective models. The literature states that hybrid methods achieve better results in comparison data sets than single models. Video summarization will be handled in two methods for online and offline approaches.

The first method is based on adding the locations of the video that were found to be unnecessary during video production to the main recorder before it was saved, and simply saving the desired locations to the main recorder. The second method is to analyze the data generated during the specific periods of the day and week after the end of the video recording and to discard the unnecessary bits.

3. ONTOLOGY DEVELOPMENT CONCEPT

The rapid development and advance of the information age, and the combination of Information and Communication Technologies in innovative gains, have resulted in the exploitation and development of many others areas, including the private sector and public institutions and enterpris. Information technologies refer to the entire range of technologies available for the monitoring, management, sharing and development of existing information. Information technology assets indicate how and how different technologies can be used for different purposes.

The information technologies used for this reason must be structured and developed in accordance with the philosophy of data integration. At this point, Information Technologies Asset Ontolog (BTVO) arises. According to some sources, the first use of ontology in computer science was in William J. Rapaport's "Philosophy of Computer Science". It was developed later and continues to be developed today. This situation, which we define as ontology development, is very new, as well as in the studies carried out in this field generally covers the after math of 2010 [3].

The ontology development method, which has come across different studies, is especially widely used in engineering studies involving the sciences. This model is used in many fields, such as providing systematic improvement over software processes, as well as providing access control, such as creating databases, integrating data, creating hierarchical classifications of systems, access control, decision support and more.

In the most common and simple definition of computer science, ontology is referred to as "the formal and clear specification of a shared conceptualization". In a more meaningful way, ontology should be revealed in a clear and understandable way and shared information should be clearly expressed by expressing semantically expressed [4].

Ontology is the most important feature that can develop when underlying the semantic web. In other words, ontology, called a smart data model, database aims to build a developable that fits the philosophy of data integration, and to build an infrastructure that is suitable for modeling. Therefore, it is expected that the ontology offered in addition to the fact that different technologies appear with different types of application is expected to serve these differences. The ontology development method also incorporates several foundations, including openness, connectivity, and conformity. An ontology developed in this perspective will gain meaningfully value through the relationship of activities and activities that arise in development processes. From a broad perspective of ontology development methods, the reflections of the software development process are clearly seen.

Many methods covering software engineering, such as analysis, design, planning and testing, have been redefined, taking into account the characteristics of ontology identification languages.

Another important point in the way of developing ontology is that the process is progressing according to a particular flow diagram, comprehensively handled, acted according to a particular order and planning. In the study to be put forward, certain researches are made and deficiencies are determined and scope, analysis and planning are determined accordingly. In accordance with the determination of these deficiencies and then the analysis and planning, the databased is modeled according to the deficiencies, the data integration is provided and the hierarchical classification is designed by looking at these studies.

The method of developing ontology is based on the conceptual framework, which is primarily developed by Gomez-Prez and later by Simpler [5]. This system, which was developed in three basic categories as the use of ontology, ontology development and support ontology, has been determined within a certain framework. The purpose of the ontology to be developed here is how the use of the RDF language to OWL language in line with the use of the target, adequacy, suitability and compliance with its purpose. In the ontology method developed, feasibility study, field analysis, conceptualization, realization, maintenance and use are determined as different situations. The main scenarios are also produced according to these situations. The conceptualization of the scenarios produced and the basic solutions through this conceptualization should be created and then these solutions should be transformed and realized to be represented. The ontology development work produced through these scenarios must be adaptable to new requirements. In other words, it is mentioned that the new information obtained after the study should be included in the process. Therefore, the systematic foundations of the study should be established by concentrating on the concept of "extensibility" and the reuse of ontology should be ensured. In this way, the new information obtained will be actively included in the system and the work will appear as a dynamic structure in terms of efficiency [5,6].

4. ONTOLOGY DEFINITION LANGUAGES

Ontology, which is a basic concept for the semantic web, contains some of its unique definition languages. On the other hand, the basis of semantic web ontology lies in facilitating the integration of internal and external information. This integration is also provided by ontology languages. Ontology studies are developed in different fields using languages, and ontogy studies developed here can be reused in other ontology studies. In addition, that it can also be connected to resources in different ontologies integrating distributed ontology studies developed here can be reused in

other ontology studies. In addition, that it can also be connected to resources in different ontologies, integrating distributed information with this connectivity.

However, ontology definition languages aim to build acknowledge model in a particular field. Two main structures play a major role in defining and developing all the standards in the Semantic Web infrastructure. The first of these is the WWW (World Wide Web) Consortium. The second is the ontology developers spread around the world. The WWW consortium constitutes the structures that form the core of the Semantic Web and that are expressed as the ontology language within these structures. There are two basic ontology language standardized; RDFS (Schema for Resource Definition Frame work) and OWL (Ontology Web Language). These two ontologies are language standardized and highly self developed languages, as we look back at the present. The following languages are following SWRL, SPARQL, RDF. RDF (Resource Definition Framework) is the most concise ontology language. Over time, RDF-S and OWL languages have risen above the RDF standards and have been defined for displaying information in a semantic web environment. The RDF language was inspired by the structure of natural language and identified three main elements to represent knowledge. These main elements are subject, object and predicate. These three elements make up the smallest pieces of information in the RDF document. These smallest pieces of information are defined by the HTTP URI standard and thus can be accessed on the web. Thanks to this method, the source RDF triplet on any server can be defined as an object and an interconnected concept network can be created on the web. Unlike RDF, RDF-S is the lightest ontology language of the consortium. The basic structure in the RDF-S ontology language is class and property definitions. With this basic structure, it is possible to define which class and which properties a concept has. In addition, new model definitions are created on top of OWL and RDF-S definitions. Another feature of RDF-S and OWL languages is that rich inferences can be made through identification logics. In other words, to put it more broadly, new information can be extracted using structures from a pre-existing knowledge model. SWRL and SPARQL languages are the other two languages defined by the consortium. SWRL (Semantic Web Rule Language) is a rule definition language. Using this rule definition language, new rule languages can be added to an existing ontology study. This definition language is mostly used in field ontology studies. SPARQL, on the other hand, is a query standard developed to query the knowledge bases of the created ontologies.

The definitions of ontology languages are given in the Ontology Development Standards Report as follows;

RDF: It is the standard that aims to define semantic information on the web and constitutes the essence of the semantic Web.

RDF-S: It is a lightweight ontology definition language. It is suitable for creating performance-efficient knowledge bases.

OWL: It is an ontology definition language with high modeling and therefore inference capabilities. It is suitable for modeling areas with semantically rich modeling requirements.

SWRL: It is the standard rule definition language. It aims to support applications that require external rule definition on the developed ontology.

SPARQL: It is the standard ontology query language in the Semantic Web environment. It aims to make all knowledge bases distributed on the semantic web questionable in a single language.

Ontology definition languages make it possible to use all the semantic information scattered over the existing website and the wealth of information revealed by the semantic web in ontology development work. When using these languages, an ontology development editor is used. The generally used ontology development editor is the program called Protégé (<http://protege.stanford.edu/>).

5. ONTOLOGY DEVELOPMENT METHODOLOGY

The working continuity of the devices that play a key role in capacity in the production facilities will be checked. Here, it is aimed to detect anomaly on the device with computer vision technology and to generate an alarm in the form of status change. Images will be collected over the cameras in the environment of the industrial establishment and stored in the computer environment. If necessary, a minimum image quality standard will be defined for capacity monitoring. Object detection will be made with the CNN model on the samples to be collected, if this model is not successful in object detection, the existing model will be trained with new samples and the success rate in object detection will be increased. After these processes at T0 time, this model will be run on new samples with regular video streaming and the difference with the previous samples will be calculated. If this difference is above a certain threshold value, the application will generate an alarm. Deep Learning method will be used to solve the problem. CNN, which is most preferred in object detection, will also be applied in this project. According to the model's success rate, the model will be retrained with new samples if necessary. If the adaptation of the pre-trained models to the relevant problem does not provide the desired success, it is aimed to propose a hybrid method combining different models. Results will be evaluated with different performance analysis metrics (Accuracy, F-Score, Sensitivity etc.). With the zero-shot method, it is aimed to classify objects without the need for labeled training data. Thanks to this approach, it will be possible to identify the machinery and equipment that are not in the training set. Meta-heuristic algorithms are considered as approaches that allow to approach

optimization problems with high success. In this context, a problem-specific model and parameter estimation approaches will be proposed.

The designs related to the tasks targeted by the image processing module are given below.

- Motion Estimation Module Design
- Design of the Identification and Identification Subsystem of Remote Object
- Design of the data summarization subsystem
- Performance analysis module design

There are processes that need to be followed when building an ontology-based subsystem. These processes are crucial to the ontology development process of the study. There are many methodologies in the literature in terms of ontology development. The methodology that we will base our development process on is the work of Noy and McGuinness in 2000. The reason for this is that the development process in the study presented supports their usable notions of ontology. Noy and McGuinness discussed this process seven steps. These steps;

1. Define the ontology scope and domain
2. Enabling ontology reuse
3. Identifying terms and term types in ontology
4. Defining classes and creating class order
5. Defining the attributes of classes
6. Defining properties of attributes
7. Defining class instances in the form

5.1. Define the ontology scope and domain. While developing an ontology, it is the first and most important step to determine the scope and domain of the ontology. After this step is determined, other steps should be taken. In other words, it should be determined what the ontology will serve to whom, how, where and how it will be used by whom and what kind of questions the developed ontology will answer in the process, and these steps should be acted upon. On the other hand, the answers to some questions may change during the development process, so there is no certainty about the answers. Although it may vary, it is important that the scope and impact area have certain limits, since ontology development is acted within certain limits.

5.2. Enabling ontology reuse. Developed ontologies should be reusable in the process. In other words, the relations between the object and the object should be in a dynamic structure and the ontology design should be done by looking at them. The main reason for this is that the remote tracking system ontology intended to be developed would refer to another tracking system ontology that would be

developed five or 10 years later. Therefore, in the relationship between an object and object, terms must be defined according to these criteria.

5.3. Identifying terms and term types in ontology. Some terms are needed to create RDF, RDFS and OWL while developing ontology. Therefore, a wide list of terms should be created. These terms emerge in line with the information obtained from the system and application to be developed. Order, attribute, data type, holistic relationship, etc., are not important when listing terms. A list is created regardless of anything else. This step is the most important step after scope and domain. Because that's where the ontology will be developed.

5.4. Terms list. The list of terms is the list that will be used to create classes. As mentioned above, everything about the system is written and the class hierarchy is built according to these kinds of things.

5.5. Defining classes and creating class order. In ontology studies, commonly common terms are taken as classes, and in other specific terms they have elements of those classes. Class patterns can also be viewed from top to bottom or bottom to top. In this way it will be easier to see the hierarchical structure of the class layout.

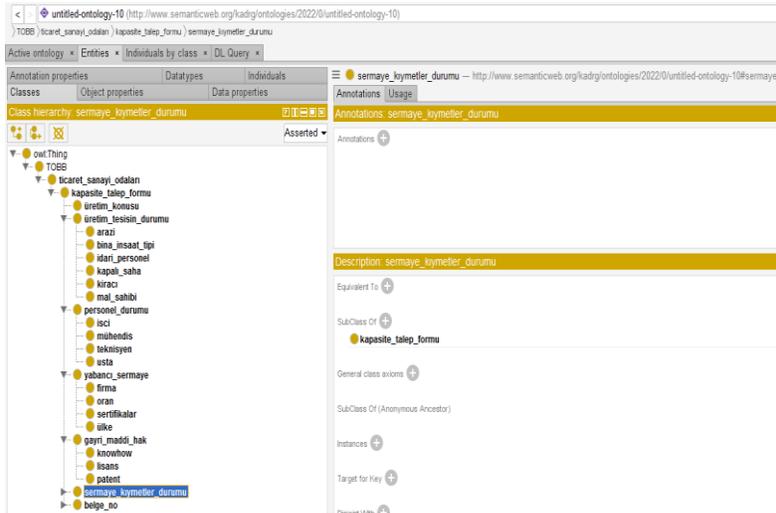


FIGURE 1. Defining classes and creating class layout.

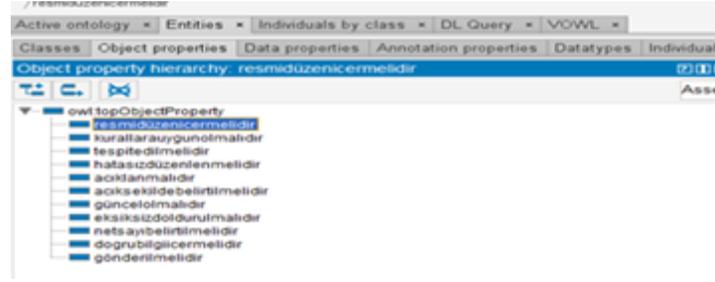


FIGURE 2. Definition of class qualities.

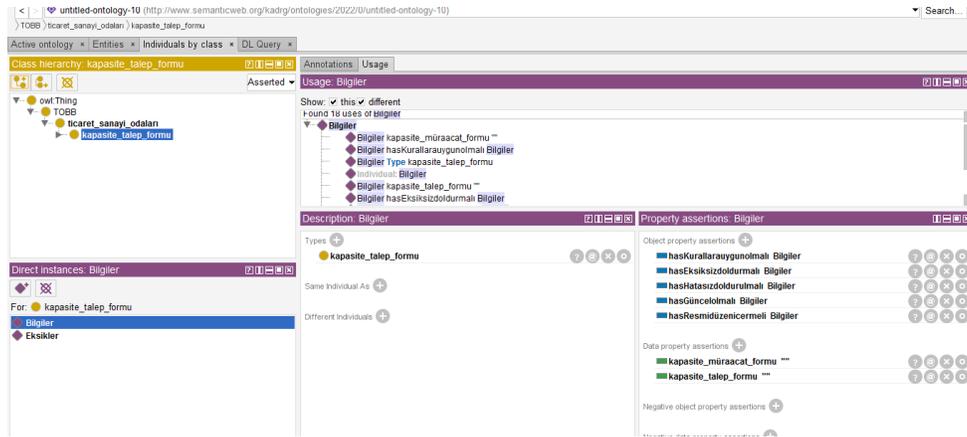


FIGURE 3. Identification of class instances.

5.6. Definition of class qualities. This title actually discussed about the internal structure of the classes, how they should be. So this is the stage at which the basic internal structures of the classes are being created Each object attribute creates the semantic relationship in the ontology study.

5.7. Identification of class instances. In the developed ontology study, class examples suitable for ontology are tried to be revealed. Both the class and object attribute properties mentioned above are the two main tools for the creation of these instances. First, it is decided which class will be created. Secondly, the instance is named and the previously determined class attribute properties are filled. This is the hardest part of the work. Because this is the part where RDF is completely revealed.

Finally, a scheme is devised using the VOWL plugin of the Protégé OWL editor to better view the organizational structure of the class and understand how the hierarchical structure is transferred to an order. This diagram provides a very clear view of which class has which subclasses, where the center point of all classes is.

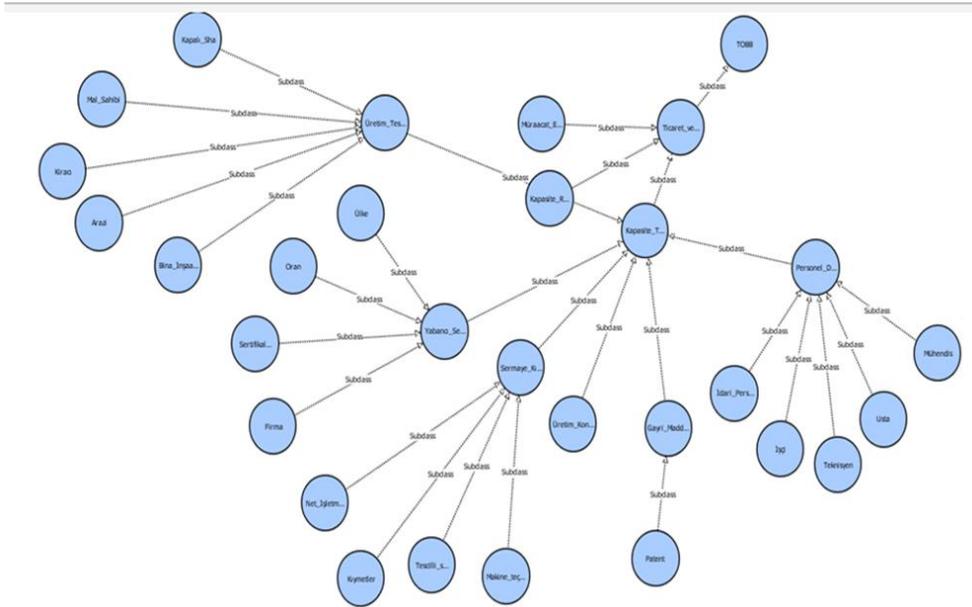


FIGURE 4. Class hierarchy scheme.

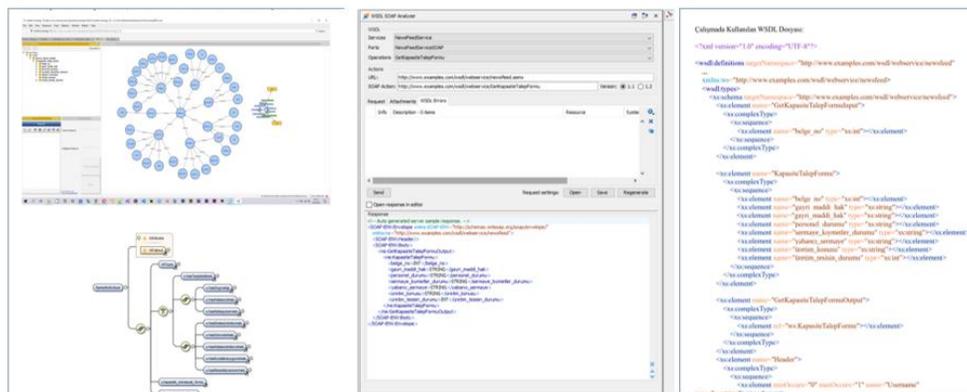


FIGURE 5. WSDL conversion and connection test work-in-progress.

6. CONCLUSIONS

When we consider information technologies, it is clearly seen that real-time monitoring is very important. In this real-time monitoring process, introducing an ontology-based system provides a serious benefit both in terms of developability and extensibility and in terms of reuse. In the proposed study, it is aimed to reveal a semantic expressiveness for the image monitoring module in remote monitoring systems.

In addition, this work is being developed by building on the architecture of deep learning. On the other hand, the method of developing ontology for the effectiveness and usability of the work has been applied. When we look at it within the scope of ontology development work, firstly the scope and domain were analyzed, and then a list of terms was created by determining the relations between the objects and objects required for reuse. By looking at the list of generated terms, classes, class properties, and class objects have been created to create the basis for extracting the RDF language that is the essence of the Semantic Web. The next process of study is still in development. While on one side data collection for image-processing continues, the ontology base is also being converted into Web Services Description Language (WSDL) format using Oxygen XML Editor.

Author Contribution Statements Corresponding author Kader Gürcüoğlu wrote about what ontology is and how it is distributed. Mehmet Güzel wrote deep learning and machine learning models of this research. Tunç Medeni and Tolga Medeni checked the research subject, content development, correct use of the ontology sub-resource and research. Halil Arslan took the task of operating the ontological database in the web service.

Declaration of Competing Interests The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement This study is carried out within the scope of the AURIZ projects supported by TEYDEP. Many thanks to TEYDEP which enabled the advancement of this project.

REFERENCES

- [1] Ünalır, O. M., Öztürk, Ö., Özacar, T., Anlamsal web için bilgi sistemi alt yapısı, TBV-BBMD, 1 (1) (2016).

- [2] Özay, A., Web servisleri ve uzak nesne tabanlı ontoloji arayüzü geliştirilmesi, (2007). Mater Thesis, Ege Üniversitesi Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı, Türkiye.
- [3] Çelebi, F. Ö., Dijital ontoloji, *Sosyal Bilimler Enstitüsü, İstanbul Medeniyet Üniversitesi, Felsefe Anabilim Dalı*, (2016).
- [4] Ünalır, O. M., Aydın, M., Aydoğan, B., Bakır, M., Durmaz, Ç., Kurtuluş, P., Bilgi teknolojileri varlık ontolojisinin geliştirilmesi, *XIV. Akademik Bilişim Konferansı Bildirileri*, (2012), 13-19.
- [5] T.C. Kalkınma Bakanlığı Bilgi Haritası Araştırma ve Geliştirme Projesi Ontoloji Oluşturma Rehberi (Şubat 2014).
- [6] T.C. Kalkınma Bakanlığı Bilgi Haritası Araştırma ve Geliştirme Projesi Ontoloji Geliştirme Yöntemi ve Standartları Raporu (Ekim 2013).

DISEASE PROGNOSIS USING MACHINE LEARNING ALGORITHMS BASED ON NEW CLINICAL DATASET

Melike COLAK¹, Talya TUMER-SIVRI², Nergis PERVAN-AKMAN¹,
Ali BERKOL¹ and Yahya EKICI³

¹Defense and Information Systems, BITES, Ankara, TÜRKİYE

²Middle East Technical University, Informatics Institute, Ankara, TÜRKİYE

³General Surgery Department, Medicana Health Point, Istanbul Beylikdüzü
International Hospital, Istanbul, TÜRKİYE

ABSTRACT. Today, artificial intelligence-based solutions are produced to facilitate human life in almost every field. The healthcare sector is one of the sectors which took advantage of these solutions. Due to reasons such as the world's ever-expanding population, ongoing epidemics, and the emergence of new disease types, it is becoming increasingly difficult for a patient to benefit from health services quickly and to make an accurate diagnosis. At this juncture, artificial intelligence reduces the patient density in hospitals, enables patients to access accurate information, and allows medical students to practice by seeing new cases. In this study, a new and reliable dataset was created with disease information obtained from various sources under the supervision of a specialist medical doctor. Then, new patient histories were added to the dataset used in the previous study, the experiments were repeated with the same algorithms, and the accuracy score comparison was presented. The created dataset includes 2006 unique patient histories, 358 symptoms, and 141 diseases and we think it will be a valuable dataset for researchers who make developments using machine learning in the field of healthcare. Various machine learning algorithms have been used in the training process to predict diseases belonging to different branches of medicine, such as diabetes, bronchial asthma, and covid. Besides, Support Vector Machine, Naive Bayes, K-Nearest Neighbors, Multilayer Perceptron, Decision Tree, and Random Forest algorithms, we also studied popular boosting algorithms such as XGBoost and LightGBM. All algorithms were validated with cross-validation and performance comparisons were made with different performance metrics such as accuracy, precision, recall, and f1-score. It is also the first study to achieve an accuracy score of 99.33% with a dataset that involves a greater number of diseases than the datasets used in the studies examined.

Keywords. Healthcare symptom checker, clinical decision support systems, machine learning.

✉ nergis.pervan@bites.com.tr-Corresponding author;  0000-0003-3241-6812

✉ melike.colak@bites.com.tr;  0000-0002-7779-4756

✉ talyatumer@gmail.com;  0000-0003-1813-5539

✉ ali.berkol@bites.com.tr;  0000-0002-3056-1226

✉ yahya.ekici@medicana.com.tr;  0000-0002-8518-8967.

1. INTRODUCTION

There are thousands of diseases that people faced all over the world, and approximately 69 million people all over the world died from various diseases or accidents in 2021 [1]. Due to the hectic pace of life, people began to pay little attention to the symptoms they felt and to not take the time to go to the hospital for a possible diagnosis of a disease. Such problems brought by the accelerating life, unfortunately, reduce the quality of life of people and shorten their lives. With the development of Artificial Intelligence (AI) and data analytics in the healthcare sector, these datasets, which store the history of thousands of patients, make sense with these technologies that produce automated approaches.

Medical organizations around the world have data on a variety of health-related topics. This data is too large for the human mind to grasp and it must be freed from noise by exploring a variety of data analytic methods to be used in various Machine Learning (ML) algorithms. Recent advances in data analytics tools and methods now enable the comprehensive use of data such as demographics, clinical diagnosis, health habits, test results, prescriptions, and service usage. As a result of processes such as data cleaning, categorization, and analysis by data analysts, this data becomes meaningful and contributes to the development of various AI studies to predict disease or identify patients at risk for other health problems. Today, cancer detection by processing X-ray images with deep learning techniques [2–5] and heart disease and stroke risk estimated with data such as heart rate and oxygen ratio in the blood [6–9]. Also, there is a study that classifies X-ray images with ML algorithms and helps orthopedists in the determination of shoulder implant types before performing revision surgery [10]. With the help of these studies, the cost required to reach health services will be reduced and the density in the hospital will be prevented. In the literature, the number of studies with ML [11–13], developed with datasets containing both categorical and image data containing the symptoms of a specific disease is quite high.

Nowadays, people want to quickly access any type of information they require from websites. When a person feels symptoms, they may be directed to false diseases unrelated to the person by searching websites containing incorrect or incomplete information. In this study, deep learning-based, ensemble-based, and tree-based approaches are presented to aim to protect people against the information pollution they will be exposed to from the internet and gives reliable results when people enter the symptoms they feel without the need to go to the hospital to get information about their disease. In this way, it is aimed to save people's time and health costs and to direct them to a medical doctor with an early diagnosis. We have seen that Support Vector Machine (SVM), Naive Bayes (NB), K-Nearest Neighbors (KNN), Random Forest (RF), and Decision Tree (DT) algorithms are frequently used in disease prediction developments. In addition to these algorithms, we tested the Extreme Gradient Boosting (XGBoost), Light Gradient Boosting Machine (LightGBM), and Multilayer Perceptron (MLP) algorithms that

we did not see used in other studies. We succeeded in surpassing the accuracy scores obtained from the SVM, random forest, decision tree, KNN, naive Bayes, and LightGBM algorithms in our first study [11]. This study contributed to the literature at two important points, under the supervision of a specialist medical doctor, a new dataset was created with disease information obtained from various sources. Secondly, it is also the first known study to reach an accuracy score of 99.33% with a dataset with a greater number of diseases than the datasets used in the studies examined.

2. LITERATURE REVIEW

Developments based on ML, which predict disease based on patient symptoms, have received a lot of attention recently due to the challenges associated with accessing healthcare services. With the DBMI dataset [14], which had 133 symptoms and 42 disease types, Gandhi et al. [15] experimented with supervised and unsupervised algorithms. In experiments with the Linear Discriminant Analysis (LDA), random forest, naive Bayes, SVM, KNN, Classification and Regression Trees (CART), and logistic regression algorithms, the accuracy score for logistic regression was the lowest of the group, coming at 80.85%. Agrawal et al. [16] suggested a new ML model in this research that combines a support vector machine and a genetic algorithm. Additionally, they attempted to reduce the number of features in the dataset, and by using their ML models, they were able to achieve adequate accuracy for all three datasets. They attained the best accuracy of 78.6% for the liver dataset consisting of categorical data. They claim that unstructured medical text data from sources including diagnoses, doctor-patient interactions, medical records, etc. would also be used in future research despite using only structured data in this study.

To get over the limitations of ML, Vinitha et al. [17], proposed leveraging big data to predict diseases based on ML. The idea is to gather information from a hospital that used the Map Reduce (MR) approach and Machine Learning Decision Tree (MLDT) algorithm to analyze data from a forum referred to as structured and unstructured data. The MR algorithm detects the possibility of disease occurrences faster than CNN-UDRP, reaching 94.8% with the standard speed. Kumar, Sharma, and Prakash [18], created a Django-based online application that uses ML algorithms to predict and provide clinical guidance for general disease, heart disease, diabetes, and liver disease. While the results of predictions for common diseases are the names of the diseases, results for predictions for specific diseases, such as heart disease, diabetes, and liver disease, are true or false. In general disease prediction, it is seen that the highest accuracy score of 90.2% among the KNN, logistic regression, random forest, and naive Bayes algorithms was achieved in random forest. The highest accuracy in heart disease was seen in logistic regression, with 92.3%. While the KNN algorithm gave the highest accuracy with 74% in the liver, it was seen that logistic regression gave the highest accuracy with 78% in diabetes. Mallela, Bhavani, and Ankaayarkanni [19], developed a GUI to get the

symptoms from the user and they used ML models such as naive Bayes and decision trees. The outputs are the disease, the accuracy of the model, its definition, and the treatment of the particular disease based on the symptoms given by the individual. This paper shows a detailed explanation of how to find the diseases from symptoms; so that the individual can contact the respective doctor of medicine and stay healthy at an early stage. A sample of 4920 patient records with diagnoses for 41 disorders was chosen by Grampurohit and Sagarnal [20] for analysis. 41 diseases made up a dependent variable. There were 132 independent variables, 95 of which were symptoms closely associated with diseases. The disease prediction system created utilizing ML techniques including decision tree, random forest, and naive Bayes is demonstrated in this research project.

Dhabarde et al. [21] use not only structured data but also textbook data, and the dataset used has 230 conditions consisting of the individual's symptoms, age, and gender. In the paper, they conducted experiments with logistic regression, naive Bayes, SVM, random forest, and decision tree algorithms, and the decision tree gave the highest accuracy score, 93.24%. Alanazi [22], proposes a method for chronic disease prediction using ML algorithms such as Convolutional Neural Network (CNN) and KNN. The proposed system used both structured and unstructured data from real life which were used for dataset preparation. The performance of the proposed model in the study shows that it is higher than the naive Bayes, decision tree, and logistic regression algorithms and provides 95% accuracy. Uddin et al. [23], conducted a study on different KNN variants (classical one, adaptive, locally adaptive, K-means clustering, fuzzy, reciprocal, ensemble, Hassanat, and generalized mean distance) and their performance comparison for disease prediction. For accuracy measurement, Hassanat KNN shows the highest average accuracy with 83.62%, followed by ensemble approach KNN with 82.34%.

For disease prediction with big data in healthcare, Joel and Priya [24], employed extended CNN. The hospital is built using this approach, which offers great accuracy, performance, and convergence speed in the medical industry. The unstructured data is employed with the CNN algorithm, which automatically selects the features, to choose a specific location and then assesses the chronic diseases that contain the structured data which extracted valuable features. The medical data and illness risk model were proposed by the innovative CNN. The suggested approach seeks to forecast the likelihood of liver-focused illness. Therefore, the hospital dataset is concerned with diseases that affect the liver, and it exclusively collects structured data from information on liver diseases. The proposed approach obtains accuracy by using disease risk modeling. Ibrahim et al. [25] proposed a method for predicting the defervescence day of fever in dengue patients using an artificial neural network. The suggested method primarily depends on clinical symptoms and indicators for detection. Data from 252 patients were collected, of which 4 patients had Dengue Fever (DF) and 248 had Dengue Haemorrhagic Fever (DHF). The neural network toolkit in MATLAB is utilized and the Multi-layer Feed-Forward Neural

Network (MFNN) technique is used in this experiment. 90% of the time, MFNN in DF and DHF correctly predicts the day of defervescence of fever. Venkatesh et al. [26], worked on five algorithms, such as random forest, KNN, naive Bayes, SVM, and decision tree; the highest accuracy score was decision tree, with 95.13%. They also have developed a user interface for patients to input their symptoms and see the disease prediction. Chauhan et al. [27], performed preprocessing on the dataset and then performed experiments on naive Bayes, decision tree, and random forest. When the experiments performed on the non-preprocessed dataset were compared with the results of the preprocessed data, it was seen that the accuracy score of the random forest was the highest in both, increasing to 95.28% in raw data and 97.64% after processing.

Maram, Kumar, and Gampala [28], stored data including 400 symptoms and 147 diseases collected from various repositories in the Hadoop Distributed File System (HDFS). Among decision trees, random forest, naive Bayes, and a new algorithm proposed in the article, the accuracy of the proposed algorithm showed the best result, with 97.60%. Through the analysis of performance measures, Ferjani [29], identifies patterns among several supervised ML model types for disease diagnosis. The supervised ML algorithms, naive Bayes, decision trees, and KNN, received the greatest attention. According to research, a support vector machine is most effective at spotting Parkinson's illness and kidney ailments. They found that the logistic regression performed well for heart disease prediction. Additionally, CNN and random forest made accurate predictions for common diseases and breast disorders, respectively. For accurate prediction, naive Bayes and KNN algorithms were used in [30] to process the person's life behaviors and check-up data. The accuracy of heart disease prediction using naive Bayes was shown to be 94.5% greater than KNN. Furthermore, compared to naive Bayes, KNN requires more memory and time. In this work, heart disease was first predicted, and then a risk prediction system using the CNN algorithm was developed to assess the risk of heart disease.

The CNN-based Multimodal Disease Prediction (CNN-MDRP) method was developed by Shirsath and Patil [31] to address the limitations of their CNN-based Unimodal Disease Prediction (CNN-UDRP) algorithm, which only analyzes structured data. In CNN-MDRP, which focuses on both structured and unstructured data, the accuracy of disease prediction is higher and faster compared to CNN-UDRP, with an accuracy score of 94.80%. Nearly 230 diseases were listed by Keniya et al. [32] with over 1000 distinctive symptoms. Various ML algorithms receive as input a person's symptoms, age, and gender. About 230 diseases were predicted using 11 different ML algorithms. The weighted KNN model had a 93.5% accuracy score, which was the highest. For disease prediction, Dahiwade, Patle, and Meshram [33] used KNN and CNN algorithms. The model accepts information from

the person's checkups and daily routine as input. With 84.5% accuracy, CNN outperforms the KNN algorithm in general disease prediction. The time and memory requirements for KNN are also higher than for CNN.

3. METHODOLOGY

3.1. Utilized Machine Learning Algorithms.

3.1.1. *K-Nearest neighbors*. KNN, which is used in both classification and regression problems, is an algorithm used in supervised learning. The basic logic is to search for K data, which are called neighbors and have the closest properties throughout the dataset, for data whose class is unknown, and assign it to the most appropriate class. There are several methods for calculating inter-data distance, the most well-known being Euclidean, Manhattan (for continuous), and Hamming distance (for categorical). The mathematical representation of the methods is shown in Eq. 1, Eq. 2 and Eq. 3:

$$\text{Euclidean Distance} = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (1)$$

$$\text{Manhattan Distance} = \sum_{i=1}^n |x_i - y_i| \quad (2)$$

$$\text{Minkowski Distance} = \left(\sum_{i=1}^n |x_i - y_i|^p \right)^{1/p} \quad (3)$$

In Eqs. 1, 2 and 3, n refers to the number of dimensions, x_i refers to the data point, and y_i refers to the new data point that is wanted to predict for all i , where $i, \in \{1, 2, \dots, n\}$, n is the size of the data points. KNN applies one of these formulas to calculate the distance between each data point and the test data. It then finds the probability that these points are similar to the test data and classifies the data according to which points share the highest probability.

3.1.2. *Support vector machine*. SVM is one of the linear supervised learning models used in classification and regression problems. When determining the class of new data, it tries to determine a dividing line (hyperplane) that best separates the available data from each other. Hyperplane can be formulated as follows,

$$f(x) = ax + c \quad (4)$$

In Eq. 4, a equals to dimensional coefficient and c equals the offset. Then, the a point closest to this line is selected from both classes, and these points are called

support vectors. The algorithm aims to provide the maximum distance difference between the support vectors and the hyperplane. There are two different types of SVM algorithms, Linear and Nonlinear. Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and the classifier is used as Linear SVM classifier. Non-linear SVM is used in cases where the dataset cannot be classified with a straight line and contains non-linear data.

3.1.3. *Naive bayes.* Naive Bayes is an algorithm based on Bayes' theorem used in classification problems, and it is known as a probabilistic classifier. It assumes that the value of probability ω_j is independent of the probability of any other event x , which means that the dependencies between the data are neglected. The simple form of calculation for Bayes' theorem is shown below.

$$P(\omega_j|x) = \frac{p(x|\omega_j) \cdot p(\omega_j)}{p(x)} \quad (5)$$

In Eq. 5, $P(\omega_j|x)$ is the posterior probability of class (target) given predictor (attribute). $p(\omega_j)$ is the prior probability of class. $p(x|\omega_j)$ is the likelihood which is the probability of the predictor given class and $p(x)$ is the prior probability of the predictor.

3.1.4. *Decision tree.* A decision tree classifier creates a model that will predict the target variable by learning the simple decision rule extracted from the feature data. In this algorithm, which has two types of nodes, decision, and leaf, the decision nodes play a decisive role in reaching the leaf, while the leaf node is the result node. Due to its nature, it is more suitable for multi-class problems. The mathematical process starts with the $D = X, y$ dataset, where each node must have a tree structure and decision rules. Each node divides the dataset into two or more discrete subsets and $D(a, b)$, in which D represents subscript (a, b) , where a is the layer number and b denotes each subset. If all tags in this subset belong to the same class, the subset is said to be pure and this node is declared as a leaf node, and this part of the tree has come to an end. Otherwise, the partitioning process continues. Data is considered pure or homogeneous if it contains only one class, and impure or heterogeneous if it contains several classes. There are various indices such as entropy and Gini to quantify the degree of impurity. Entropy is the amount of information required to accurately describe some samples. That is, if the sample is homogeneous, all elements are similar with entropy 0, otherwise if the sample entropy is divided evenly by a maximum of 1. The other index, the Gini index, is defined as a measure of inequality in the data and has a value between 0 and 1. If the Gini index is 0, the data is considered completely homogeneous and all elements are similar. A Gini index of 1 means the maximum inequality between the elements. It is the sum of the squared probabilities of each class. Mathematical expressions for the Gini index and entropy are shown in Eq. 6 and Eq. 7.

$$\text{Gini index} = 1 - \sum_{i=1}^n (p_i)^2 \quad (6)$$

where p_i is the probability of an object being classified to a particular class.

$$\text{Entropy} = \sum_{i=1}^c -p_i \log_2 p_i \quad (7)$$

where p denotes the probability.

3.1.5. *Random forest.* Random forest, one of the supervised learning algorithms, combines multiple classifiers to solve a classification or regression problem and improve the model's performance. Instead of relying on a single decision tree to reach the result, it combines the output of multiple decision trees to obtain safer results. High variance and low bias are characteristics of decision trees, and by averaging decision trees, the variance component of the model is reduced. It is possible to create the unknown samples by averaging the prediction with Eq. 8 and Eq. 9,

$$I = \frac{1}{N} \sum_{n=1}^N f(x) \quad (8)$$

$$\sigma = \sqrt{\frac{\sum_{n=1}^N (f(x) - \hat{f})^2}{N - 1}} \quad (9)$$

Where σ denotes the uncertainty and N denotes the sample number.

Hence, random forest is a bagging algorithm, which is a method of generating different training subsets from training data with replacement. A final result is reached by calculating the feature importance in each decision tree that makes up the random forest. Feature importance is calculated as the reduction in node impurity weighted by the probability of reaching that node. The node probability can be calculated by dividing the number of samples reaching the node by the total number of samples. The higher the value, the more important the feature. For each decision tree, a node's importance is calculated using Gini importance. Then the importance of each feature on a decision tree is then calculated and these can then be normalized to a value between 0 and 1 by dividing by the sum of all feature importance values. At the random forest level final feature importance is its average over all the trees. The sum of the feature's importance value on each tree is calculated and divided by the total number of trees. The feature importance describes which features are relevant and it sometimes leads to model improvements by employing the feature selection.

3.1.6. *Extreme gradient boosting.* XGBoost is a specially optimized version of the gradient boosting algorithm for high performance and speed and uses decision trees as “weak” predictors. Since it contains many parameters that can be optimized, it provides the opportunity to improve the model. In addition to system optimizations such as parallel operation, tree pruning, and hardware optimization, algorithmic improvements such as regularization, cross-validation, weighted quantile sketch, and sparsity-aware split make XGBoost a more efficient algorithm compared to other models. Mathematically in the XGBoost algorithm, the objective function is shown in Eq. 10.

$$O(t) = \sum_{i=0}^n Q(y_i, y'^{t-1} + f_t(x_i)) + K \quad (10)$$

Normalization function can be defined as

$$Nor(f_t) = \kappa T + 0.5\lambda \sum_{i=0}^T W_j^2 \quad (11)$$

κ =Controlling factor for the leaf node number, T =Leaf node number, W_j =Weight-age of the j leaf nodes, λ =Overfitting controlling factor, K =Constant, in Eq. 10 and Eq. 11.

XGBoost performs exceptionally well on structured tabular data rather than data like images and audio.

3.1.7. *Light gradient boosting machine.* LightGBM is a gradient-assisted decision tree algorithm that is very similar to XGBoost, increasing the model’s accuracy and reducing memory usage. LightGBM develops models that have lower error rates and learn faster by using a leaf-oriented strategy instead of a level-oriented strategy in decision trees. According to the [34] where the model is introduced, it has been concluded that LightGBM is 20 times faster than other models. The decision tree growth process used by XGBoost and LightGBM differs significantly. Decision trees in XGBoost grow horizontally with a method called level-wise, but decision trees in LightGBM grow vertically with a leaf-wise approach. The leaf-wise approach is an effective method as it makes LightGBM an efficient technique for high-dimensional datasets. XGBoost and LightGBM decision tree growth processes are shown in Figure 1. LightGBM’s multithreaded optimization and leaf growth technique with depth restriction help to reduce excessive XGBoost memory usage so that big data processing can be done more quickly, with fewer false alarms, and with lower missed detections.

3.1.8. *Multilayer perceptron.* Multilayer perceptron is mainly used in recognition, prediction, regression, and pattern classification and it is the most basic type of feed-forward network architecture, compared to other major types. Here, the units are arranged into a set of layers, and each layer contains some number of identical units. The network is considered fully connected when each unit in a layer connects to each unit in the layer above it. The input layer is the top layer, and its units use

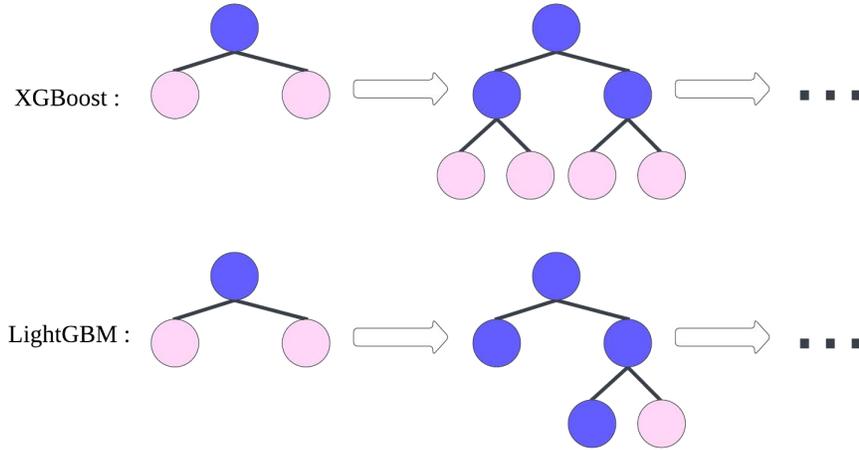


FIGURE 1. XGboost and LightGBM decision tree growth processes.

the values of the input properties as input. For each value produced by the network, there is one unit in the output layer in case of regression or binary classification, and K units in classification cases involving K classes. All the levels in between are called hidden layers because it is not known what these units have to calculate in advance and must learn this while training. Input units, output units, and hidden units are the names of units located at these levels, respectively. Backpropagation MLPs can be used to solve problems that cannot be classified linearly, such as the XOR problem in sensors.

3.2. Dataset. In our previous study [11], the dataset prepared by DBMI [14] was used. The previous dataset consisted of 133 symptoms and 42 diseases and included 306 patient histories. In this study, we add disease and symptom data from various sources under the supervision of a specialist medical doctor; and use a new dataset in which we increased the total number of cases approximately 14 times with data augmentation.

3.2.1. Data collection. The diseases in the dataset were searched on the Internet and matched with the relevant symptoms. After this process, a dataset containing 150 diseases and 383 symptoms was created. The created dataset was checked by a medical doctor, and if there was incorrect symptom information, it was removed from the dataset and the reliability of the dataset was ensured. After making sure that the data were correct, the dataset was ready for preprocessing with 141 diseases and 358 symptoms. A part of the dataset created at the end of this stage is shown in Table 1.

TABLE 1. Raw dataset.

disease	symptom 1	symptom 2	symptom 3	...
acne	thirst	blackheads	skin rash	...
acute	pancreatitis	vomiting	diarrhoea	...
addison's disease	fatigue	lethargy	low mood	...
adenovirus	diarrhoea	cough	runny nose	...
aids	extra marital contacts	patches in throat	high fever	...
...

3.2.2. *Preprocessing.* Any manipulation applied to raw data is called data preprocessing. Through this process, scattered data becomes organized, and problem-appropriate, and transforms into a format that can be processed effectively in ML developments. To provide reliable, accurate, and robust findings for enterprise applications, practically every sort of data analysis, data science, or ML development requires some kind of data preprocessing. At this stage, the columns in the dataset were changed to consist of 358 symptom names and target columns, and the rows to consist of 141 patient cases. Data preprocessing was carried out so that the value at the intersection of the row and column is “1” if the disease contains the relevant symptom, and “0” if it does not. Except for the target column values, disease, which is a textual type that is converted to numerical form using a label encoder, all of the column values in the dataset are numbers. A part of the dataset formed after preprocessing is shown as an example in Table 2.

TABLE 2. Preprocessed dataset.

...	yellow urine	yellowish eyes	yellowish skin	disease
...	0	0	0	acne
...	0	0	0	acute
...	0	0	0	addison's disease
...	0	0	0	adenovirus
...	0	0	0	aids
...

3.2.3. *Data augmentation.* By creating additional data points from existing data, a group of techniques known as data augmentation can be used to artificially enhance the amount of data. This includes making minor adjustments to the data or creating new data points using ML models. By creating additional and distinct instances for training datasets, data augmentation helps ML models perform better and more accurately with large datasets. For this reason, it is aimed to obtain more realistic results by adding new data to the dataset used in our previous study [11] and applying data augmentation. Some symptoms play a key role in the prediction of

diseases. An example of this is the symptom of loss of smell for Covid disease. For this reason, while creating a new patient history with data augmentation, attention was paid to including these symptoms in each patient's history. In addition, the data augmentation process is aimed to make the data more suitable for real-world cases by completely randomly determining how many times each disease will increase with this process and what symptoms it will contain. As a result of this process, the dataset was increased approximately 15 times, and a new dataset was created with 2006 patient histories.

3.2.4. Data splitting. When starting an ML project, one of the first considerations to be discussed is how to use existing data. Typically referred to as training and test sets, dividing data into two groups is a standard strategy. When making predictions on data that was not used to train the model, ML algorithms perform as predicted using the train-test separation process. The training set is used for estimating parameters, comparing models, and all other activities required to arrive at a final model. The test set is used to predict a final, unbiased assessment of the model's performance only at the end of these activities. Since the test set is used to measure the performance of the model after the training is over, it reveals a high but erroneous performance result that the model had seen before. The most common technique for splitting the dataset into training and test sets is random splitting. The used dataset, which was randomly divided by 85-15%, 1713 samples were determined as training data and 303 samples as test data. The cross-validation method was utilized in the study in addition to the traditional method of creating test data at random to evaluate the model's performance.

3.3. Training. In the previous study, k-fold cross-validation was used with the k value chosen as 5 to avoid overfitting, and the average accuracy of the classifier was taken. The choice of k is usually 5 or 10, but there is no formal rule. As k gets larger, the difference in size between the training set and the re-sampling subsets gets smaller. As this difference decreases, the bias of the technique becomes smaller [35]. K-fold cross-validation is not suitable for evaluating unbalanced classifiers because the data is divided into k-folds with a uniform probability distribution. This method may work for data with a well-balanced class distribution, but it is dangerous when the distribution is severely dispersed, with one or more of the folds having few or no samples from the minority class. It means that most model evaluations produce misleading results, as the model only needs to predict the majority class correctly. Accuracy scores with high deviations at the end of the training k times help to understand that the data in the dataset is not evenly distributed [36]. In this study, the k value remained constant to make a reliable comparison of the change in performance metrics with the expansion of the dataset.

In the experiments, the hyperparameter optimization library hyperopt was used to find the parameters showing the highest accuracy score, and the learning rate in the LightGBM model was determined as 0.1175, max bin 316, max depth 3, num

leaves 200, objective parameters binary. In the XGBoost model, hyperparameters are determined as one drop true, learning rate 0.3, colsample by tree 0.5698, gamma 0.5296, max depth, and objective multi:softprob. In random forest, criterion parameter is entropy, max-depth is 14,056 with and the n-estimators parameter is 340. In SVM, the C parameter was determined as 1.144, gamma, 0.278 and kernel rbf. In KNN, we determined the n-neighbors parameter value as 3. The decision tree's criterion value is determined by Gini and in naive Bayes, the alpha parameter is 1.0. As a result of the experiments, the highest accuracy score was achieved in the MLP model by using the activation function tanh, hidden layer sizes 32 and max iter 3000 hyperparameters. For other classifier parameters not mentioned above, the default parameter values of the scikit-learn library are based.

4. RESULTS

In the experiments conducted in the study, the lowest accuracy score was obtained as 92% among all algorithms. The fact that this value was 79.3% in the previous study, clearly showed the positive effect of the data augmentation techniques we applied to the disease detection experiments. Among all algorithms, it was seen that the highest scores in all metrics were obtained in the ensemble learning method, the random forest algorithm, which consists of many decision trees. The most important feature of the random forest algorithm is that it can work with high performance in regression and classification problems with datasets containing continuous or categorical variables. The decision trees that made up the random forests are prone to errors in classification problems with a large number of classes and relatively few training examples. The decision tree, which had the lowest accuracy score of 79.3% in the first dataset containing a small number of samples from each class, increased to 95.37% as a result of increasing the number of data belonging to each class. However, SVM appears to perform as well as random forest in its measurements across all metrics, achieving the second-highest precision, recall, and F1-score results. It also appears that the SVM achieves the highest average accuracy in the k-fold algorithm. In addition to the remarkable performance of random forest and SVM, we observed that the MLP algorithm was the 3rd highest-performing algorithm in the experiments. Although the lightGBM algorithm has a high accuracy score compared to the studies in the literature, it gave the lowest result among the algorithms in all performance metrics in the experiments conducted in our study. The most valuable output of the experimental results is that we obtained higher accuracy scores in SVM, lightGBM, random forest, naive Bayes, decision tree, and KNN algorithms compared to our previous study.

All algorithms were fitted using the hyperparameters mentioned in the training section; where estimators were evaluated using k-fold cross-validation where k is 5 and average accuracy scores are shown in Table 3. Precision, recall, and f1-score performance measures were used to examine the performance of the models. The blue values in Table 3 show the best results for that metric among the models,

while the red color indicates the worst result. The model results are shown in Table 3 with the specified performance measures and methods, and the comparison of accuracy scores of the methods is shown in Figure 2.

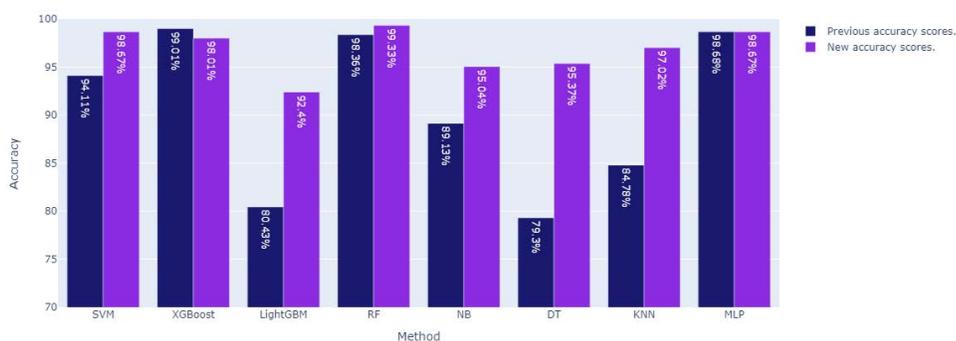


FIGURE 2. Comparison of accuracy scores of the utilized methods.

TABLE 3. Model results with the specified performance measures and methods.

Method	5-Fold Average Accuracy	Accuracy	Precision	Recall	F1-Score
SVM	99.25	98.67	99.02	98.81	98.92
XGBoost	97.83	98.01	97.47	96.37	96.92
LightGBM	88.39	92.40	90.80	90.56	89.72
RF	99.20	99.33	99.51	99.30	99.40
NB	93.50	95.04	92.79	93.70	93.24
DT	93.55	95.37	96.08	95.55	95.81
KNN	98.31	97.02	97.52	96.76	97.14
MLP	99.05	98.67	98.48	97.86	98.17

5. CONCLUSION

Technological developments in healthcare are highly important as they are directly related to human life. In this paper, a disease prediction study is developed where people can obtain free, reliable, and fast information about their health. The development we offer serves important purposes such as reducing the patient density in hospitals and early diagnosis of viral diseases. In addition, with our study, which we developed under the supervision of a specialist medical doctor, it is possible to

prevent people from obtaining information about their health from websites that contain information pollution. As an alternative to the dataset used in our previous study, a new dataset consisting of clinical data was created under the supervision of medical doctors, and the data size was increased. The new study includes combinations of different types of symptoms and diseases. The highest accuracy score is reached with the random forest classifier at 99.33%, while the lightGBM has the lowest accuracy score among the algorithms with 92.40%. Although our study consisted of a high number of disease classes compared to the studies in the literature, all algorithms obtained reliable results with an accuracy score of over 92%. For future studies, the determination of the severity of the symptoms in the dataset by the doctors and the learning of the model during the training process, taking into consideration the symptom weights of each disease, are among our further studies.

Author Contribution Statements The authors jointly worked on the study. All authors read and approved the final copy of the manuscript.

Declaration of Competing Interests The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement In this study, we made progress from our previous work [11] in terms of both increasing the number of data and improving the algorithms used. The study that forms the basis of this paper was presented at the 5. International Conference on Theoretical and Applied Computer Science and Engineering (ICTACSE).

REFERENCES

- [1] Our World in Data, (2022). Available: <https://ourworldindata.org/births-and-deaths/>. [Accessed: December 2022].
- [2] Cantalay, P. J., Uçan, O. N., Zontul, M., Diagnosis of breast cancer from X-ray images using deep learning methods, *J. Ponte*, 77 (6), (2021), <https://doi.org/10.21506/j.ponte.2021.6.1>.
- [3] Wang, Y., Yang, F., Zhang, J., Yue, X., Liu, S., Application of artificial intelligence based on deep learning in breast cancer screening and imaging diagnosis, *Neural Comput. & Applic.*, (2021), 9637–9647, <https://doi.org/10.1007/s00521-021-05728-x>.
- [4] Mobark, N., Hamad, S., Rida, S. Z., CoroNet: Deep neural network-based end-to-end training for breast cancer diagnosis, *Appl. Sci.*, 12 (14), (2022), 7080, <https://doi.org/10.3390/app12147080>.
- [5] Manishkumar, S. H. and Saranya, P., Detection and classification of breast cancer from mammogram images using adaptive deep learning technique, *2022 6th Int'l Conf. on Dev., Circ. & Syst.*, (2022), 327-331, <https://doi.org/10.1109/ICDCS54290.2022.9780770>.
- [6] Reddy, K. V. V., Elamvazuthi, I., Aziz, A. A., Paramasivam, S., Chua, H. N., Pranavanand, S., Heart disease risk prediction using machine learning classifiers with attribute evaluators, *Appl. Sci.*, 11 (18) (2021), 8352, <https://doi.org/10.3390/app11188352>.
- [7] Bharti, R., Khamparia, A., Shabaz, M., Dhiman, G., Pande, S., Singh, P., Prediction of heart disease using a combination of machine learning and deep learning, *Comp. Intell. & Neurosci.*, (2021), 8387680, <https://doi.org/10.1155/2021/8387680>.

- [8] Mehmood, A., Iqbal, M., Mehmood, Z. et al., Prediction of heart disease using deep convolutional neural networks, *Arab. J. for Sci. & Eng.*, 46 (2021), 3409–3422, <https://doi.org/10.1007/s13369-020-05105-1>.
- [9] Puri, H., Chaudhary, J., Raghavendra, K. R., Mantri, R., Bingi, K., Prediction of heart stroke using support vector machine algorithm, *21 8th Int'l Conf. on Sm. Comput. & Comm.*, (2021), 21-26, <https://doi.org/10.1109/ICSCC51209.2021.9528241>.
- [10] Sivari, E., Güzel, M. S., Bostanci, E., Mishra, A., A novel hybrid machine learning based system to classify shoulder implant manufacturers, *Healthcare*, (2022), 10, 580, <https://doi.org/10.3390/healthcare10030580>.
- [11] Çolak, M., Sivri, T. T., Akman, N. P., Berkol, A., Ekici, Y., A study of disease prediction on weighted symptom data using deep learning and machine Learning algorithms, *Int'l Conf. on Theor. & Appl. Comput. Sci. & Eng.*, (2022), 116-119, <https://doi.org/10.1109/ICTACSE50438.2022.10009857>.
- [12] Xie, S., Yu, Z., Lv, Z., Multi-disease prediction based on deep learning: A survey, *Comput. Mod. in Eng. & Sci.*, 128 (2) (2021), 489-522, <https://doi.org/10.32604/cmes.2021.016728>.
- [13] Ahsan, M., Siddique, Z., Machine learning-based heart disease diagnosis: A systematic literature review, *Artif. Intel. in Med.*, 128 (2022), <https://doi.org/10.1016/j.artmed.2022.102289>.
- [14] Disease-Symptom Knowledge Database, (2022). Available: <https://people.dbmi.columbia.edu/>. [Accessed: September 2022].
- [15] Gandhi, K., Mittal, M., Gupta, N. and Dhall, S., Disease prediction using machine learning, *Int'l J. for Res. in Appl. Sci. & Eng. Tech.*, 8 (2020), 500-507, <http://doi.org/10.22214/ijraset.2020.6077>.
- [16] Agrawal, A., Agrawal, H., Shivam, M., Sharma, M., Disease prediction using machine learning, *Proc. of 3rd Int. Conf. on IoT & Connected Tech.*, (2018), <http://dx.doi.org/10.2139/ssrn.3167431>.
- [17] Vinitha, S., Sweetlin, S., Vinusha, H., Sajini, S., Disease prediction using machine learning over big data, *Comput. Sci. & Eng.: An Int'l J.*, 8 (1) (2018), <http://dx.doi.org/10.2139/ssrn.3458775>.
- [18] Kumar, A., Sharma, G. K. and Prakash, U. M., Disease prediction and doctor recommendation system using machine learning approaches, *Int'l J. for Res. in Appl. Sci. & Eng. Tech.*, 9 (2021), 34-44, <https://doi.org/10.22214/ijraset.2021.36234>.
- [19] Mallela, R. C., Bhavani, R. L., Ankayarkanni, B., Disease prediction using machine learning techniques, *2021 5th Int. Conf. on Trends in Electronics & Informatics*, (2021), 962-966, <https://doi.org/10.1109/ICOEI51242.2021.9453078>.
- [20] Grampurohit, S., Sagarnal, C., Disease prediction using machine learning algorithms, *2020 Int. Conf. for Emerging Tech.*, (2020), 1-7, <https://doi.org/10.1109/INCET49848.2020.9154130>.
- [21] Dhabarde, S., Mahajan, R., Mishra, S., Chaudhari, S., Manelu, S., Shelke, N. S., Disease prediction using machine learning algorithms, *Int. Res. J. of Mod. in Eng. Tech. & Sci.*, 4 (2022), 379-384.
- [22] Alanazi, R., Identification and prediction of chronic diseases using machine learning approach, *J. of Healthc. Eng.*, 2022 (2022), 1-9, <https://doi.org/10.1155/2022/2826127>.
- [23] Uddin, S., Haque, I., Lu, H., Moni, M. A., Gide, E., Comparative performance analysis of K-nearest neighbor (KNN) algorithm and its different variants for disease prediction, *Sci. Rep.*, 12 (2022), 1-11, <https://doi.org/10.1038/s41598-022-10358-x>.
- [24] Joel, G. N., Priya, S. M., Improved ant colony on feature selection and weighted ensemble to neural network based multimodal disease risk prediction (WENN-MDRP) classifier for disease prediction over big data, *Int. J. of Eng. & Tech.*, 7 (2018), 56-61, <https://doi.org/10.14419/ijet.v7i3.27.17654>.

- [25] Ibrahim, F., Taib, M. N., Abas, W. A., Guan, C. C., Sulaiman, S., A novel dengue fever (DF) and dengue haemorrhagic fever (DHF) analysis using artificial neural network (ANN), *Comput. Methods Programs Biomed.*, 79 (3) (2005), 273-281, <https://doi.org/10.1016/j.cmpb.2005.04.002>.
- [26] Venkatesh, K., Dhyanes, K., Prathyusha, M. and Teja, C. H. N. , Identification of disease prediction based on symptoms using machine learning, *JAC: J. Comp. Theory*, 14 (2021), 86-93, <https://doi.org/10.1155/2022/2826127>.
- [27] Chauhan, R. H., Naik, D. N., Halpati, R. A., Patel, S. J. and Prajapati, A. D., Disease prediction using machine learning, *Int. Res. J. of Eng. & Tech.*, 7 (2020), 2000-2002.
- [28] Maram, B., Kumar, K. S. and Gampala, V., Symptoms based disease prediction using bigdata analytics, *Turk. J. of Phys. Ther. & Rehab.*, 32 (3) (2021), 3228-3234.
- [29] Ferjani, M., Disease prediction using machine learning, *Bournemouth Univ.*, (2020), <http://dx.doi.org/10.13140/RG.2.2.18279.47521>.
- [30] Awari, S. V., Diseases prediction model using machine learning technique, *Int. J. of Sci. Res. in Sci. & Tech.*, 8 (2) (2021), 461-467, <https://doi.org/10.32628/IJSRST>.
- [31] Shirsath, S. S., Patil, S., Disease prediction using machine learning over big data, *Int. J. of Innov. Res. in Sci. Eng. & Tech.*, 7 (6) (2018), 6752-6757, <https://doi.org/10.15680/IJIRSET.2018.0706059>.
- [32] Keniya, R., Khakharia, A., Shah, V., Gada, V., Manjalkar, R., Thaker, T., Warang, M. and Mehendale, N., Disease prediction from various symptoms using machine learning, *SSRN Electron. J.*, (2020), <https://dx.doi.org/10.2139/ssrn.3661426>.
- [33] Dahiwade, D., Patle, G., Meshram, E., Designing disease prediction model using machine learning approach, *3rd Int. Conf. on Comput. Methodol. & Comm.*, (2019), 1211-1215, <https://doi.org/10.1109/ICCMC.2019.8819782>.
- [34] Ke, G., Meng, Q., Finley, T., Wang, T. , Chen, W., Ma, W., Ye, Q., Liu, T., LightGBM: A highly efficient gradient boosting decision tree, *Adv. in Neural Inf. Process. Syst.*, (30) (2017), 3146-3154.
- [35] Kuhn, M., Kjell, J., Applied Predictive Modeling, Springer, New York, 2013.
- [36] He, H., Ma, Y., Imbalanced Learning: Foundations, Algorithms, and Applications, Willey, 2013, 188, <http://dx.doi.org/10.1002/9781118646106>.

MILITARY CAMOUFLAGE CLASSIFICATION WITH MASK R-CNN ALGORITHM

Ilkay KARATEPE¹ and Vasif NABIYEV¹

¹Computer Engineering Department, Karadeniz Technical University,
Trabzon, TÜRKİYE

ABSTRACT. Camouflage, which is used as an art of hiding by living things in nature, started to be used in the military field in the 19th century with the widespread use of long-range firearms. When factors such as different nations, environment and climate are considered, we come across camouflages in various colors and patterns. Over time, the camouflage patterns adopted and used by countries or unions have become national identity. This study is on the classification and segmentation of camouflaged soldiers of 5 countries with deep learning. While the similarity of the camouflaged area with the background makes segmentation difficult, it becomes difficult to classify each camouflage pattern due to the cut of the fabric and the different locations of the pattern pieces on each soldier. There are different studies in the literature that are referred to as camouflage or pattern classification. The mentioned studies are in the form of segmentation of camouflaged object or classification of camouflaged objects of different types. Since the segmented and classified objects in this study are camouflaged soldiers, what is expected from the deep learning algorithm is to classify the objects mainly according to the camouflage pattern, not their outlines. In the study, 861 camouflaged soldier images were collected for 5 countries (Türkiye-Azerbaijan, USA, Russia, China, France) and polygonal labeling was made. Türkiye and Azerbaijan are considered a class as they have similar camouflages. For the solution of the problem, military camouflage classification was discussed with the Mask R-CNN algorithm, which is widely used today for object detection, segmentation and classification, and the importance of deep learning algorithms has been proven with such a difficult problem. The training resulted in 0.005219 classification loss and 0.03985 masking loss. The classification and segmentation success rate of the study is 95%.

Keywords: Camouflage, deep learning, Mask R-CNN, classification, segmentation.

 ilkaykaratepe@gmail.com-Corresponding author;  0000-0002-6627-1503
 vasif@ktu.edu.tr;  0000-0003-0314-8134.

1. INTRODUCTION

Until the mid-1800s, armies dressed their soldiers in bright clothes [1]. With the development of more accurate firearms, the first military camouflage was used in 1848 as yellowish dull uniforms by a regiment of the British Indian Army [2]. Towards the end of the 19th century, plain khaki-colored uniforms were adopted by the British army [3]. In addition, in the First World War, patterned uniforms consisting of different colors, which they called 'Fragmentation Camouflage', were used in the German army [3]. While the design of camouflage patterns was in the hands of designers and artists at the beginning of the 20th century, computer technology began to be used in the field of camouflage in the late 1970s [3]. Today, many countries have adopted different camouflage patterns. Even if the camouflages used have different patterns, it is very difficult to derive the mathematical model of these patterns for classification. Although there are image processing approaches for detecting, segmenting or classifying an object, these problems can be solved more accurately and quickly with the evolution of deep learning models.

Different studies have been conducted in the literature on camouflage or pattern classification. Doğan et al. compared different deep learning algorithms such as AlexNet, Vgg16, Vgg19, ResNet50 and GoogleNet on classifying leaf patterns [4]. Gupta et al., with the 58-layer Convolutional Neural Network (CNN) model they customized, made a study that detects various soldiers and equipment with foggy, snowy, night vision consisting of 22 classes [5]. Bayram et al. extracted textural features for camouflage images using Local Binary Patterns (LBP) and classified them with Artificial Neural Networks (ANN), Support Vector Machines (SVM) and K-Nearest Neighborhood Algorithm (KNN) [6]. In this study, Mask R-CNN algorithm [7], which was developed for regional-based object detection and segmentation, was used for segmentation and classification of camouflaged soldier images belonging to 5 different countries. Ömeroğlu et al. used the Mask R-CNN algorithm to detect hangars on high-resolution satellite images with an average accuracy of 85% [8]. Amri et al. used YOLOv5 [9] and Mask R-CNN [10] for stadium detection on multispectral images.

2. MATERIAL AND METHOD

2.1. Material. For this study, camouflaged soldier images belonging to 5 countries (Türkiye-Azerbaijan, USA, Russia, China, France) were collected and used as a data set. Türkiye and Azerbaijan have similar camouflages due to their joint military efforts. For this reason, it was chosen as a class. The total number of photographs is 558 and contains a total of 861 images of soldiers. The photos are set to 700x700 dimensions and randomly allocated for 80% (448) training and 20% (110) testing.

Images are labeled and ready for training with the LabelMe [11] tool. The number of soldier images used in Table 1 is given separately for each country. Examples of camouflage fabrics belonging to 5 countries are given in Figure 1 and examples of images divided into 5 classes used in Figure 2.

TABLE 1. Distribution of images used in the study.

	Training	Test
Türkiye-Azerbaijan	130	28
USA	120	26
Russia	130	34
China	182	35
France	141	35
Total	703	158



FIGURE 1. Examples of camouflage fabrics: Türkiye-Azerbaijan [12], USA [13], Russia [14], China [15], France [16].



FIGURE 2. Examples of camouflaged soldiers: Türkiye-Azerbaijan [17], USA [18], Russia [19], China [20], France [16].

2.2. **Method.** In this study, images were made ready for education by labeling them polygonally with the LabelMe tool. Labeled images were trained with Mask R-CNN algorithm in 20000 iterations and instance segmentation was successfully provided for 5 classes. A fixed learning rate (lr) was not used during the training. The learning rate was reduced in certain steps with the gamma (γ) factor, which is

known as the decay or reduction of the learning rate in the deep learning literature, and the training process was continued until the total loss took constant values. Using a fixed and large learning rate will cause the total loss rate to decrease rapidly but will cause the local minimum point to be skipped after a while. Using a fixed and small learning rate will prolong the learning process and may lead to overfitting. Considering the extremely similar characteristics of the classes in the problem, a decreasing learning step approach was preferred. The effect of the gamma multiplier is shown in figure 3. In the study, the learning rate started with the value of 0.004 and was updated by multiplying the value of 0.95 γ so that it decreases by 5% for every 500 steps. After the first 10000 steps of the training, the gamma was set to 0.99 and an update was made every 200 steps.

The Mask R-CNN algorithm used in the study is a region-based convolutional neural network developed based on Faster R-CNN [22]. Instance segmentation is done with Mask R-CNN. This masking process is also the identification of target object pixels in the bounding box predicted by Faster R-CNN.

In Faster R-CNN, first feature mapping is performed with convolutional neural network and then regions containing objects are determined (Region Proposal Network-RPN). The operation in RPN is the basic CNN operation. The region that receives the IoU (Intersection over Union) value, which is the ratio of the highest intersection to the union, from the regions predicted by RPN is sent to the intersection regions called ROI (Region of Interest) in the network. Here, since the bounding box sizes from the RPN are variable, it is converted to a fixed size by clipping (ROI pooling), which the classifier will work more accurately. After the size transformation in the ROI, the class of the predicted region is also determined. The results obtained in Faster R-CNN are sent to the fully connected layer and the class and bounding box information is generated.

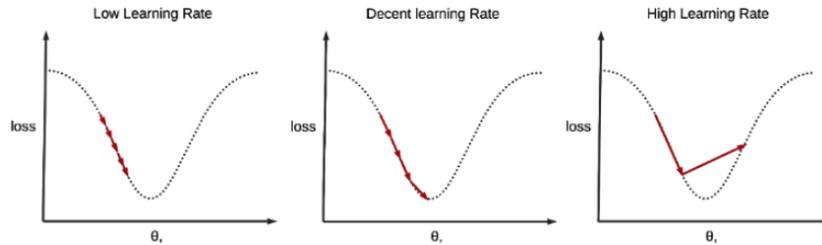


FIGURE 3. Effect of gamma multiplier on learning process: low-constant learning rate, decreasing learning rate, high-constant learning rate [21].

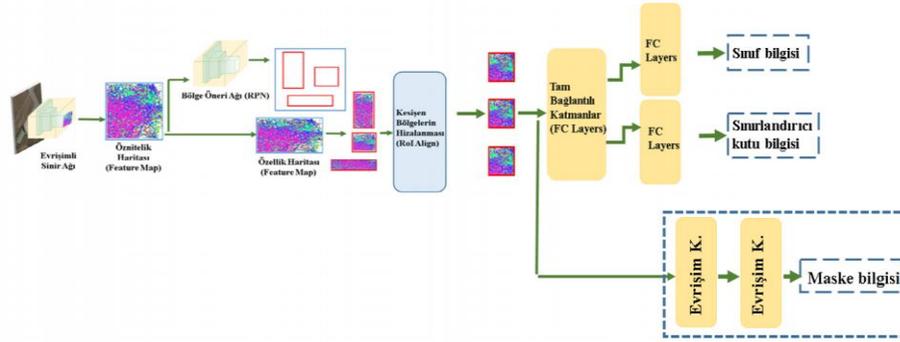


FIGURE 4. General structure of Mask R-CNN [8].

In Mask R-CNN, after ROI, in parallel with the FC layer, mask information is generated by semantic classification. The general structure of Mask R-CNN is given in figure 4. In Mask R-CNN, the loss function (L) is the sum of classification losses (L_{cls}), masking losses (L_{mask}) and frame losses (L_{box}) [23]. The loss function is shown in Equation (1).

$$L = L_{cls} + L_{mask} + L_{box} \quad (1)$$

3. EXPERIMENTAL STUDIES

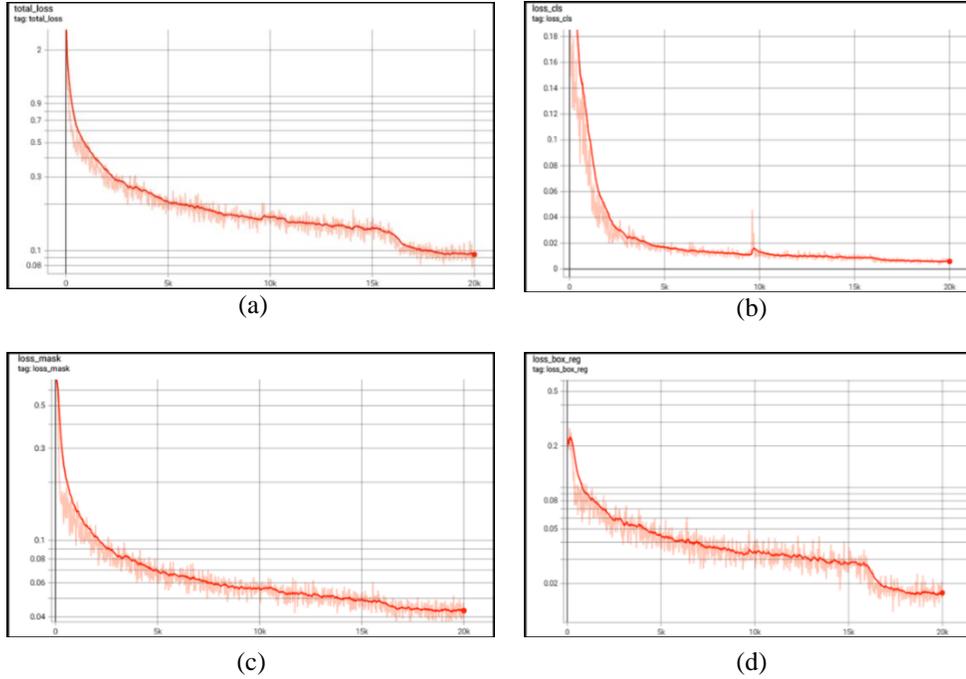
The study was carried out with Google Colab on a computer with Intel(R) Xeon(R) CPU @ 2.20GHz processor, Tesla T4 graphics card and 26 GB primary memory, with Detectron2 [24] developed by Facebook Artificial Intelligence Research (FAIR). For training, 558 photographs consisting of 5 classes were collected. This data set contains a total of 861 camouflaged soldiers. In the data set, each class was randomly divided into 80% training and 20% testing.

4. RESULTS

When the model was applied to the test images, it was seen that it predicted with an average of 95% accuracy. The loss rates achieved with 20000 iterations are shown in Table 2. The training metrics formed at the end of the training are given in Figure 5 and the segmentation results obtained with the test images are given in Figure 6.

TABLE 2. Loss rates.

Total Loss	Loss of Classification	Loss of Masking	Loss of Frame
0.061919	0.005219	0.03985	0.01685

FIGURE 5. Training metrics: (a) total loss, (b) loss of classification, (c) loss of *masking*, (d) loss of *frame*.

5. CONCLUSIONS

Considering the studies in the literature, even if image processing methods give results for the segmentation of camouflaged objects, different methods are additionally preferred for the selection of some threshold values. In the camouflaged object classifications, since the classified objects are different from each other, the general outlines of the objects rather than the pattern are an effective factor in classification. Considering the mathematical complexity of military camouflage patterns, it is a difficult problem to classify with image processing methods. The fact

that the segmented objects in our study were only soldiers enabled the algorithm used to base the patterns for classification. Looking at the classification loss from the training loss rates given in Table 2, it is seen that Mask R-CNN is successful in solving this complex classification problem. On the other hand, the average accuracy estimation is a good value for CNN algorithms.

The results revealed that CNN algorithms are effective not only for classification of objects with different outlines, but also for classification problems with high mathematical complexity. In further studies, different studies can be done by increasing the number of classes.

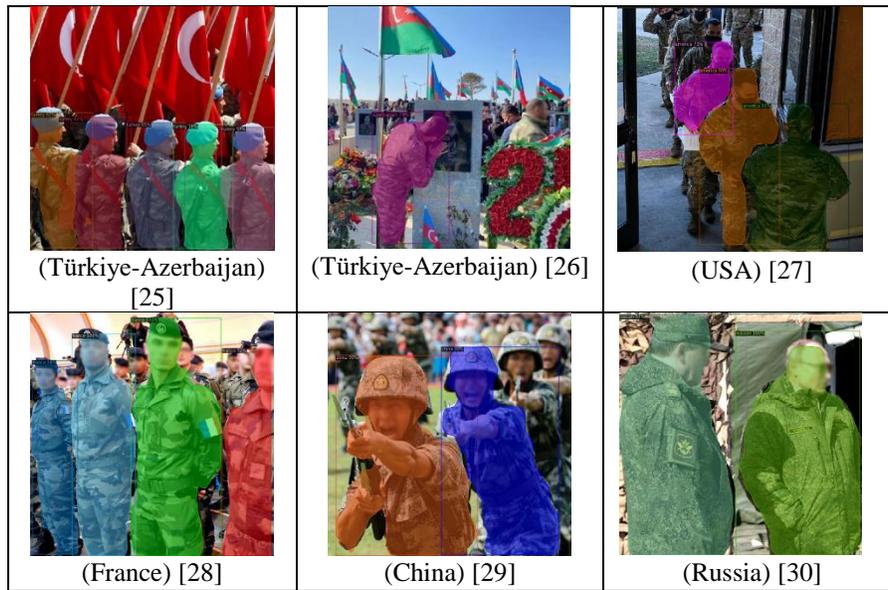


FIGURE 6. Sample segmentation results with Mask R-CNN.

Author Contribution Statements The authors contributed equally to this work.

Declaration of Competing Interests The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement We thank Seda KARATEPE for her help in the preparation of the data set used in the study.

REFERENCES

- [1] Talas, L., Baddeley, R., Cuthill, I., Cultural evolution of military camouflage, *Philos. Trans. R. Soc. B: Biol. Sci.*, 372 (2017), 20160351, <https://doi.org/10.1098/rstb.2016.0351>.
- [2] Hodson-Pressinger, S., Khaki uniform, 1848-1849: first introduction by Lumsden and Hodson, *J. Soc. Army Hist. Res.*, 82 (2004), 341–347.
- [3] Singh, R., Study of Graphic Camouflage Patterns on Battle Uniform and Improving the Pattern Used by Indian Army, (2014). Master Thesis, PDPM Indian Institute of Information Technology, Design and Manufacturing Jabalpur, Jabalpur.
- [4] Doğan, F., Türkoğlu, I., Derin öğrenme algoritmalarının yaprak sınıflandırma başarımlarının karşılaştırılması, *SAUCIS*, 1 (2018), 10-21.
- [5] Gupta, A. and Gupta, U., Military surveillance with deep convolutional neural network, *International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT)*, (2018), 1147-1152, <https://doi.org/10.1109/ICEECCOT43722.2018.9001381>.
- [6] Bayram, E. and Nabyev, V., Classification of camouflage images using local binary patterns (LBP), *29th Signal Processing and Communications Applications Conference (SIU)*, (2021), 1-4, <https://doi.org/10.1109/SIU53274.2021.9478040>.
- [7] He, K., Gkioxari, G., Dollár, P., and Girshick, R., Mask R-CNN, *IEEE International Conference on Computer Vision (ICCV)*, (2017), 2980-2988, <https://doi.org/10.1109/ICCV.2017.322>.
- [8] Ömeroğlu, A. N., Kumbasar, N., Oral, E. A. and Ozbek, I. Y., Mask R-CNN algoritması ile hangar tespiti hangar detection with Mask R-CNN algorithm, *27th Signal Processing and Communications Applications Conference (SIU)*, (2019), 1-4, <https://doi.org/10.1109/SIU.2019.8806552>.
- [9] Redmon, J., Divvala, S., Girshick, R. and Farhadi, A., You only look once: unified, real-time object detection, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2016), 779-788, <https://doi.org/10.1109/CVPR.2016.91>.
- [10] Amri, M. B., Yedjour, D., El Amin Larabi, M. and Bakhti, K., Stadium detection from Alsat-2 and Google-Earth multispectral images using YOLOv5 and Mask R-CNN, *4th International Conference on Pattern Analysis and Intelligent Systems (PAIS)*, (2022), 1-4, <https://doi.org/10.1109/PAIS56586.2022.9946887>.
- [11] Torralba, A., Russell, B. C. and Yuen, J., LabelMe: online image annotation and applications, *Proceedings of the IEEE*, 98 (8) (2010), 1467-1484, <https://doi.org/10.1109/JPROC.2010.2050290>.
- [12] GIJOE SURPLUS, Turkish Army Digital Camo Uniform - Combat Shirt, (2021). Available at: <https://gijoebcn.com/en/turkey/2427-18712-turkey-army-digital-camo-combat-shirt-used-condition.html>. [Accessed: March 2023].

- [13] The New York Times, Would you consider serving in the U.S. Armed Forces?, (2020). Available at: <https://www.nytimes.com/2020/01/15/learning/would-you-consider-serving-in-the-us-armed-forces.html>. [Accessed: March 2023].
- [14] War on the Rocks, How is the Russian military responding to covid-19?, (2020). Available at: <https://warontherocks.com/2020/05/how-is-the-russian-military-responding-to-covid-19/>. [Accessed: March 2023].
- [15] South China Morning Post, South China Sea stand-off: Vietnam takes hard line while negotiating code of conduct but will Beijing listen?, (2018). Available at: <https://www.scmp.com/news/asia/diplomacy/article/2180074/south-china-sea-stand-vietnam-takes-hard-line-while-negotiating>. [Accessed: March 2023].
- [16] U.S. Army medical research and development command, First French Army Soldier Working at the ISR, (2012). Available at: https://mrdc.health.mil/index.cfm/media/articles/2012/first_french_army_soldier_working_at_the_ISR. [Accessed: March 2023].
- [17] Wikipedia, Two Turkish soldiers salute in Kabul, (2013). Available at: [https://tr.wikipedia.org/wiki/Dosya:Two_Turkish_soldiers_salute_in_Kabul_\(4699277363\).jpg](https://tr.wikipedia.org/wiki/Dosya:Two_Turkish_soldiers_salute_in_Kabul_(4699277363).jpg). [Accessed: March 2023].
- [18] Defense Visual Information Distribution Service, Brig. Gen. Moses Kaoiwi Jr. receives farewell gift, (2021). Available at: <https://www.dvidshub.net/image/6890790/brig-gen-moses-kaoiwi-jr-receives-farewell-gift>. [Accessed: March 2023].
- [19] 7sur7, Les troupes russes n'auraient pas enregistré d'avancées majeures ces dernières 24h, (2022). Available at: <https://www.7sur7.be/monde/les-troupes-russes-n-auraient-pas-enregistre-d-avancees-majeures-ces-dernieres-24h~ae288a77/>. [Accessed: March 2023].
- [20] English.gov.cn, The state council the people's republic of China, President Xi reviews navy in South China Sea, (2018). Available at: http://english.www.gov.cn/news/top_news/2018/04/13/content_281476110326046.htm. [Accessed: March 2023].
- [21] Deep Learning Wizard, Learning rate scheduling (2019). Available at: https://www.deeplearningwizard.com/deep_learning/boosting_models_pytorch/lr_scheduling/. [Accessed: March 2023].
- [22] Ren, S., He, K., Girshick, R. and Sun, J., Faster R-CNN: towards real-time object detection with region proposal networks, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39 (6) (2017), 1137-1149, <https://doi.org/10.1109/TPAMI.2016.2577031>.
- [23] Yayla, R. and Şen, B., Region-based segmentation of terrain fields in SAR images, *28th Signal Processing and Communications Applications Conference (SIU)*, (2020), 1-4, <https://doi.org/10.1109/SIU49456.2020.9302432>.
- [24] Wu, Y., Kirillov, A., Massa, F., Lo, W. Y. and Girshick, R., Detectron2, July 2022.

- [25] Haber7, Türkiye'nin hangi ülkelerde askeri üssü var?, (2021). Available at: <https://www.haber7.com/foto-galeri/66858-turkiyenin-hangi-ulkelerde-askeri-ussu-var>. [Accessed: March 2023].
- [26] The Brooklyn News, Brooklyn Times – Azerbaijan marks anniversary of victory in war, (2021). Available at: <https://bklynnews.com/front-pages/front-pages-november-2021/the-brooklyn-times-750-am-11-22-2021/>. [Accessed: March 2023].
- [27] The Mike Church Show World, Tuesday Red Pill Diaries-New D-Day For U.S. Armed Forces & All Murican Gubbmint: CoronaDoom, (2021). Available at: https://mikechurch.com/wp-content/uploads/2021/08/New_DDAY_1920.jpg. [Accessed: March 2023].
- [28] 24 Канал, The French army is preparing for hostilities: the country has unprecedented military exercises, (2023). Available at: https://24tv.ua/ru/vo-francii-startovali-masshtabnye-voennye-ucheniya-24-kanal_n2263123. [Accessed: March 2023].
- [29] Foreign Policy Magazine, China's untested military could be a force—or a flop, (2018). Available at: <https://foreignpolicy.com/2018/11/27/chinas-untested-military-could-be-a-force-or-a-flop/>. [Accessed: March 2023].
- [30] News.ru, The head of the General Staff told Putin an anecdote, (2019). Available at: <https://news.ru/russia/glava-genshtaba-rasskazal-putinu-anekdot/>. [Accessed: March 2023].

INSTRUCTIONS TO CONTRIBUTORS

Communications Faculty of Sciences University of Ankara Series A2-A3: Physical Sciences and Engineering is a single-blind peer reviewed open access journal which has been published since 1948 by Ankara University, accepts original research articles written in English in the fields of Physics, Engineering Physics, Electronics/Computer Engineering, Astronomy and Geophysics. Review articles written by eminent scientists can also be invited by the Editor.

Article-processing charges: The publication costs for Communications Faculty of Sciences University of Ankara Series A2-A3: Physical Sciences and Engineering are covered by the journal, so authors do not need to pay an article-processing and submission charges. The PDF copies of accepted papers are free of charges and can be downloaded from the website. Hard copies of the paper, if required, are due to be charged for the amount of which is determined by the administration each year.

Submission: All manuscripts should be submitted via our online submission: <https://dergipark.org.tr/en/journal/2457/submission/step/manuscript/new> Note that only two submissions per author per year will be considered. Once a paper is submitted to our journal, all co-authors need to wait 6 months from the submission date before submitting another paper.

Cover Letter: Manuscripts should be submitted in the PDF form used in the peer-review process together with **THE COVER LETTER** and the source file (Supporting File). In the cover letter the authors should suggest the most appropriate Area Editor for the manuscript and potential four reviewers with full names, universities and institutional email addresses. Proposed reviewers must be experienced researchers in your area of research and at least two of them should be from different countries. In addition, proposed reviewers must not be co-authors, advisors, students, etc. of the authors. In the cover letter, the author may enter the name of anyone who he/she would prefer not to review the manuscript, with detailed explanation of the reason. Note that the editorial office may not use these nominations, but this may help to speed up the selection of appropriate reviewers.

Preparing your manuscript: Manuscripts should be typeset using as DOC or LaTeX. Authors will submit their manuscript and the cover letter via our submission system. A template of manuscript can be reviewed in <https://dergipark.org.tr/tr/download/journal-file/20554> (or can be reviewed in [pdf form](#)).

Title Page: The title page should contain the title of the paper, full names of the authors, affiliations addresses and e-mail addresses of all authors. Authors are also required to submit their Open Researcher and Contributor ID (ORCID)'s which can be obtained from <http://orcid.org> as their URL address in the format <http://orcid.org/xxxx-xxxx-xxxx-xxxx>. Please indicate the corresponding author.

Abstract and Keywords: The abstract should state briefly the purpose of the research. The length of the Abstract should be between 50 to 5000 characters. At least 3 keywords are required.

Math Formulae: Formulas should be numbered consecutively in the parentheses ().

Tables: All tables must have numbers (TABLE 1) consecutively in accordance with their appearance in the text and a legend above the table. Please submit tables as editable text not as images.

Figures: All figures must have numbers (FIGURE 1) consecutively in accordance with their appearance in the text and a caption (not on the figure itself) below the figure. Please submit figures as EPS, PDF, TIFF or JPEG format.

Authors Contribution Statement, Declaration of Competing Interests and Acknowledgements should be given at the end of the article before the references.

References: The following format for the references should be used. Authors are urged to use the Communication.csl style (<https://dergipark.org.tr/en/download/journal-file/18514>) in Mendeley Desktop or Zotero automated bibliography. If manual entry is preferred for bibliography, then all citations must be listed in the references part and vice versa. Below, It has no relationship with the text, but can be used to show sample citations such as; for articles [1, 4], for books/booklets/theses [3], and for proceedings/conferences etc. [2].

[1] Demirci, E., Unal, A., Özalp, N., A fractional order SEIR model with density dependent death rate, Hacettepe J. Math. Stat., 40 (2) (2011), 287–295.

[2] Gairola, A. R., Deepmala, Mishra, L. N., Rate of approximation by finite iterates of q-Durrmeyer operators, Proc. Natl. Acad. Sci. India Sect. A Phys. Sci., 86 (2) (2016), 229–234.

[3] Lehmann, E. L., Casella, G., Theory of Point Estimation, Springer, New York, 2003.

[4] Özalp, N., Demirci, E., A fractional order SEIR model with vertical transmission, Math. Comput. Model., 54 (1-2) (2011), 1–6, <https://dx.doi.org/10.1016/j.mcm.2010.12.051>.

Peer-review policy: The Editor may seek the advice of two, or three referees, depending on the response of the referees, chosen in consultation with appropriate members of the Editorial Board, from among experts in the field of specialization of the paper. The reviewing process is conducted in strict confidence and the identity of a referee is not disclosed to the authors at any point since we use a single-blind peer review process.

Copyright: Copyright on any open access article in Communications Faculty of Sciences University of Ankara Series A2-A3: Physical Sciences and Engineering is licensed under a Creative Commons Attribution 4.0 International License (CC BY).

Declarations/Ethics With the submission of the manuscript authors declare that:

- All authors of the submitted research paper have directly participated in the planning, execution, or analysis of study;
- All authors of the paper have read and approved the final version submitted;
- The contents of the manuscript have not been submitted, copyrighted or published elsewhere and the visual-graphical materials such as photograph, drawing, picture, and document within the article do not have any copyright issue;
- The contents of the manuscript will not be copyrighted, submitted, or published elsewhere, while acceptance by the Journal is under consideration.
- The article is clean in terms of 'plagiarism', and the legal and ethical responsibility of the article belong to the author(s). Author(s) also accept that the manuscript may go through plagiarism check using IThenticate software;
- The objectivity and transparency in research, and the principles of ethical and professional conduct have been followed. Authors have also declared that they have no potential conflict of interest (financial or non-financial), and their research does not involve any human participants and/or animals.

Archiving: Research papers published in Communications Faculty of Sciences University of Ankara are archived in the Library of Ankara University (Volume 1-63) and Dergipark immediately following publication with no embargo.

Editor in Chief

Commun. Fac. Sci. Univ. Ank. Ser. A2-A3.

Ankara University, Faculty of Sciences

06100 Beşevler, ANKARA – TÜRKİYE

C O M M U N I C A T I O N S

FACULTY OF SCIENCES
UNIVERSITY OF ANKARA

DE LA FACULTE DES SCIENCES
DE L'UNIVERSITE D'ANKARA

Series A2-A3: Physical Sciences and Engineering

Volume: 65

Number: 1

Year: 2023

Research Articles

Ö.F. YANIK, H.A. ILGIN, A comprehensive computational cost analysis for state-of-the-art visual slam methods for autonomous mapping	1
K. AÇICI, Ö.M. ERDAL, A. YILMAZ, M. UNAL, G.E. BOSTANCI, M.S. GÜZEL, Airborne imagery and lidar based 3D reconstruction using commercial drones.....	16
F. EKİNCİ, Ionization and phonon production by ¹⁰ B ions in radiotherapy applications.....	30
K. GÜRCÜOĞLU, T.D. MEDENİ, İ.T. MEDENİ, M.S. GÜZEL, H. ARSLAN, Ontology development for web services to be used within the scope of remote monitoring project	38
M. ÇOLAK, T.T. SİVRİ, N.P. AKMAN, A. BERKOL, Y. EKİCİ, Disease prognosis using machine learning algorithms based on new clinical dataset.....	52
İ. KARATEPE, V. NABİYEYEV, Military camouflage classification with Mask R-CNN algorithm.....	69