



Journal of INNOVATIVE SCIENCE and ENGINEERING

Volume 7
Issue 2
Year 2023

E-ISSN: 2602-4217

www.jise.btu.edu.tr

A Laboratory Study on the Design and Performance Evaluation of Pitot-Tube

Yasemin ESKİ¹ , Babak VAHEDDOOST¹ , Damla YILMAZ^{1*} 

¹ Bursa Technical University, Department of Civil Engineering, Bursa, Turkey

Cite this paper as: Eski Y., Vaheddoost B. and Yilmaz D. (2023). A Laboratory Study on the Design and Performance Evaluation of Pitot-Tube, 7(2):122-132.

*Corresponding author: Damla YILMAZ
E-mail: damla.yilmaz@btu.edu.tr

Received Date: 13/01/2023
Accepted Date: 17/08/2023
© Copyright 2023 by
Bursa Technical University. Available
online at <http://jise.btu.edu.tr/>



The works published in the journal of Innovative Science and Engineering (JISE) are licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Abstract

Due to the increasing demand for water resources worldwide, this commodity and its spatial and temporal properties are of the interest for decision makers and scientists. On the other hand, the accuracy in detecting the physical characteristics of the water flow such as velocity is among the most important aspects of the hydraulic studies. The pitot tube, which is not widely used in the open channel hydraulic practices, is one of the equipment used for determination of the flow velocity. In this study, we have addressed the design, fabrication, and laboratory experiments related to a pitot tube to investigate its applicability for open channel experiments. A 3D-printed pitot-tube is designed and used in a set of experiments carried out in an open channel, with different flow rates (three experiments). As a result, the relative error rates were interpreted by comparing the velocity rates obtained with the help of the water level difference in the differential manometer (V_m) and the velocity rates obtained from the flow continuity equation in the open channel (V_o). Results indicated a 50% bias, while the scatter analysis showed that the associated deviations match a linear equation and once used in the interpretation of the results, the linear transformation reveals a 3% bias in the experiments.

Keywords: 3D-printer, Flow velocity, Open channel hydraulics, Pitot- tube design

1. Introduction

The sustainability of water and water resources in general is important in evaluation and exploitation of the roadmaps of the modern societies. Due to the increasing demand for the fresh water resources, conducting studies for determination of the flow characteristics such as the velocity, flow rate, surface water profiles, and the water depth seems inevitable. As a result, the continuous investigation of such parameters has an important role both in examining the behavior of the flow and in the design and planning of the hydraulic structures. The velocity of flow, is one of the most important characteristics of the flow, and a key parameter in determining the erosion, sediment movement, the permanency of the flow, the flow conditions (subcritical or supercritical), and the amount of flow passing through a section. So far, the previous laboratory and field studies on the transmission capacity and velocity of the water in open channels have revealed valuable information about the basic principles of the phenomena [1-7]. Along with the most well-known theoretical methods such as continuity, Manning's, and Bernoulli's equations; alternative methods/equipment such as current meter (e.g. hydrometric reel), orifice-meter, Particle Image Velocimetry (PIV) or laser aided measurements, and Acoustic Doppler Velocity Meter (ADV) are used in previous studies [8]. For instant, current meter (sometimes called as Moline) has a propeller that rotates with the movement of water. The flow rate is usually calculated afterward, by taking into account the number of revolutions during the rotation of the propeller [9]. On the other hand, Orifices are simple instruments used to measure the flow rate under constant load in water tanks or pipes. Hence, the flow velocity can be obtained theoretically with the help of Bernoulli equation. Alternatively, the ADV is a device that measures and records the water flow velocity via emitted sound waves at different depths, that hits the particles in the water, and returns to the ADV for the evaluation [10]. Pitot-tube is also an alternative equipment for measuring the velocity in a fluid (gas or liquids), while its application is mostly limited to the gases. This device with intertwined pipe system, takes the difference between the dynamic and hydrostatic head. Afterward, with the help of Bernoulli's equation, velocity of the flow can be acquired. In this context, previous studies showed that in practice the application of Pitot-tube in measurement of velocity provides consistent results [11-15]. In a study conducted by Ghaznawi [16], an open channel of 0.055 (width) \times 1.44 (length) m was used. In the conducted experiments, velocity rates were measured using a fabricated Pitot-tube, with a ruler and a medical infusion set. The velocity and pressure distributions were calculated both numerically and experimentally, and compared with the water depths and velocity rates obtained by the experiments. Afterward, the velocity profiles were examined and the differences were determined between the experimental and numerical results in the regions close to the water surface at the upstream. When the Normalized Root Mean Square Error (NRMSE) results calculated for the velocity profiles in all experiments were examined, it was concluded that the differences decreased with the increase of the flow rate. When the velocity measurements taken on the applied weir were examined, it was concluded that the experimental and numerical results were more consistent at the upstream region. As a result, an agreement between the experimental and numerical results was obtained and therefore, the application of the fabricated Pitot tube was suggested for further application. In the study conducted by Demirel [17], the most suitable model was investigated for the determination of hydropower potential in small river basins, while the velocity measurements were carried out using floats and a Pitot-tube method. In the performed field measurements of small rivers without the gauging stations, the cross-sections considered in the float method were also measured with the help of Pitot-tube. Afterward, the velocity rates obtained by floats and Pitot-tube methods

were evaluated, and the instantaneous flow rates were estimated in three-month periods representing the precipitation regimes of eight streams that do not have a gauging station. It was concluded that the difference between the flow rates measured with the Pitot-tube and the float method was quite small and negligible.

According to the brief statement of the art detailed above, and due to the continues need for confirmation or confrontation of the idea, this study aimed to address the application of the Pitot-tube once again for open channel flow practices. To achieve this, a Pitot-tube is designed and fabricated to be used in a set of laboratory experiments. Initially, the Pitot-tube is designed and fabricated using a 3D printer (i.e. stereolithography based). Then, a differential manometer is used in order to read the dynamic and hydrostatic heads of the flow measured by the Pitot-tube. Alternatively, the flow velocity determined using the flow depth before the forefront of the Pitot-tube, later to be used in continuity equation. Finally, the velocity rates obtained by continuity equation and the fabricated Pitot-tube are compared with each other to confirm the degree of concordance between the data sets. Hence, the following sections addresses the theoretical aspect and background of Pitot-tube and differential Manometer equipment later to be followed by the 3D design, fabrication, lab experiments and result section.

1.1. Pitot Tube

Pitot-tube is made up of two nested pipes, that is used in measuring the velocity of fluids. While the outer pipe measures the static pressure (in here hydrostatic head), the interior pipe is for measuring the dynamic pressure caused by the movement of the fluid [18]. The outlets of these pipes, are then connected to a differential manometer, that measures the height, h to determine the point velocity of the flow with the help of well-known $V=\sqrt{2gh}$ equation. Pitot tubes can be designed and fabricated with consideration to the basic principles given in the statement of the art. According to Berkun [19], the diameter of the outer pipe in the Pitot-tube should be considered twice the diameter of the inner pipe, and seven holes must be provided to measure the static pressure (Fig. 1). The other must, is the distance of the provided holes to the forefront of the Pitot-tube, that is six times the diameter of the outer pipe ($6D$), while the distance from the Pitot tube anchor (forefront) to the center of the perpendicular tip (elbow) should be 14 times of the outer pipe diameter ($14D$).

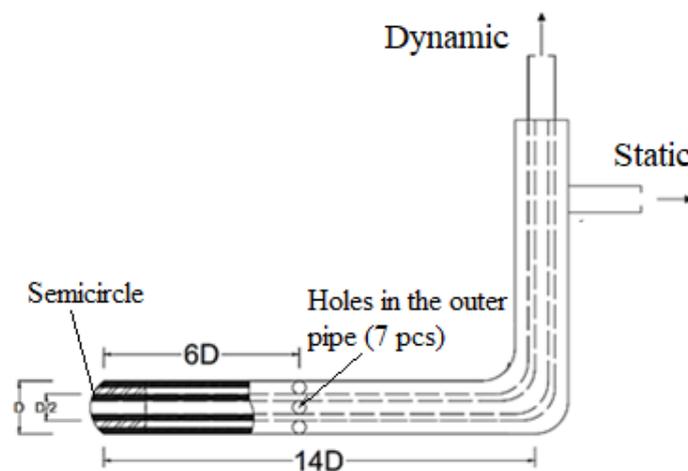


Figure 1. Pitot-tube impact tip and design principles [19]

1.2. Differential Manometer

A differential manometer is usually used to measure the pressure differences between two desired points with the help of a U-shaped pipe filled up with a liquid of a known specific gravity. In this respect, it can be used to measure the pressure gradient between the dynamic and the hydrostatic heads of a flow at a specific location [20]. Therefore, in practice a differential manometer is generally preferred in measurement of a specific fluid (Fig. 2) that provides eligible reads.

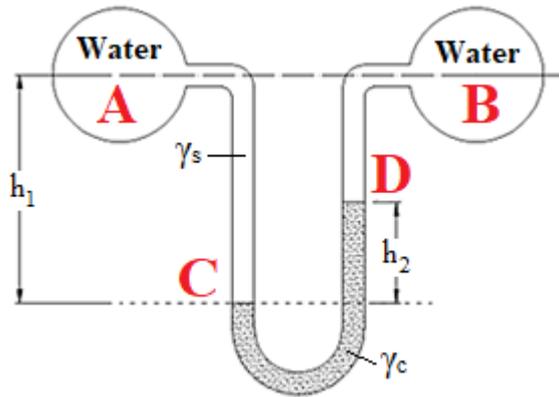


Figure 2. Differential manometer

Consequently, in the present study a differential manometer is connected to the fabricated Pitot-tube to measure the head gradient between the pipes. The pressure difference between the pipes can be calculated as,

$$P_A + h_1\gamma_s = h_2\gamma_c + (h_1 - h_2)\gamma_s + P_B \quad (1)$$

Therefore, the pressure difference (ΔP) between point A and B can be determined as,

$$\Delta P = P_A - P_B = h_2(\gamma_c - \gamma_s) \quad (2)$$

In these equations, γ_c , γ_s , h_1 , h_2 , P_A and P_B are the specific weight of the liquid in the manometer (dyed water is used in this study), the specific weight of the fluid (water in the channel), the manometer head in point A, the manometer head in point B, the manometer pressure in point A, and the pressure of manometer in point B respectively. Afterward, the difference between point A and B could be calculated using h_2 and the difference between the specific weights of the liquid in the manometer and the containers [19].

2. Materials and Methods

The design, fabrication, and experiments of this study are all carried out in the Hydraulics Laboratory of Civil Engineering Department at Bursa Technical University located in Bursa province of Turkey. In this respect, the following sections addresses, the design and fabrication of the Pitot-tube and the conducted open channel experiments at the laboratory.

2.1. Pitot tube design

In the first stage, an 8 mm diameter (D) is selected for the fabrication of the Pitot-tube by examining the commercially available samples. Based on this reference value, the inner diameter of the pitot-tube is determined as 4 mm (D/2); while the distance of the holes from the forefront of the tube is determined as 48 mm (6D). Additionally, and along with these criteria, the length of the tip parallel to the current is considered to be 112 mm

(14D). The outlet of the pipes is then fabricated at the same length, parallel to the direction of the upstream flow (Fig. 3a).

The Fusion 360 software is then used for the 3D sketching and designing of the Pitot-tube with respect to the dimensions detailed before (Fig. 3b). The designed Pitot-tube is then sliced with 0.05 mm resolution (i.e. defining thickness and number of layers for 3D printing) and made ready for 3D printing with the help of Anycubic Photon Workshop 64 software, commercially available for purchasing the Anycubic Photon M3 Max 3D printer (available at the laboratory). As a printer that uses stereolithography (SLA) rather than Fused Deposition Modeling (FDM), a UV-sensitive liquid Resin with $1.05 \sim 1.25 \text{ g/cm}^3$ density and $150 \text{ m Pa}\cdot\text{s}$ viscosity is used in fabrication of the pitot-tube. Once exposed to the UV lights of the SAL machine, it solidifies and fabricates the solid object as expected. Hence, relatively adequate number of supports are used in bottom and sides of the Pitot-tube for protection of the 3D printed object (i.e. pitot-tube) and avoid fabrication errors (Fig. 3c). The 3D printing fabrication with $\pm 0.05 \text{ mm}$ precision, is then initiated and successfully lasted after 1.5 hrs (6 cm/hr in average). Afterward, in order to clean the excess liquid resin and enhance the 3D printing quality, the fabricated Pitot-tube is washed using Isopropyl Alcohol ($\text{C}_3\text{H}_8\text{O}$) and normal tap water (H_2O); and then cured with the UV rays with the help of Anycubic Wash and Cure Plus machine. Finally, an air compressor is used to remove the excess liquid remains inside of the Pitot-tube, and the final installation and assembly between Pitot-tube and differential manometer is achieved.

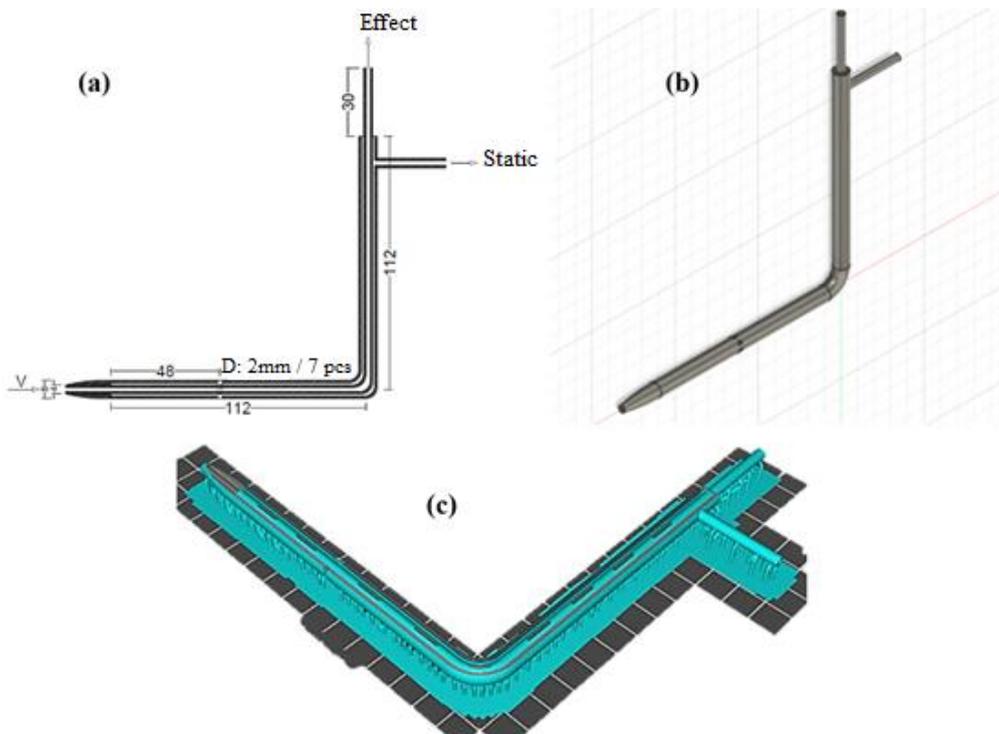


Figure 3. Pitot-tube (a) designed dimensions, (b) Fusion 360 design, and (c) prepared object in Anycubic Photon Workshop 64

2.2. Open Channel Experiments

As detailed before, initially the Pitot-tube and differential manometer are connected with each other (Fig. 4a). The water head reads of the differential manometer were made readable by dying the water with red food coloring. An open channel with $200 \times 10 \times 9 \text{ cm}$ dimensions is then used in the experiments, while a cross-section located at the 120 cm (A relatively distance from the upstream gate and the honeycomb (to regulate the flow) that assumed to be long enough to reach a stationary state) from the upstream is used for conducting measurements (Fig. 5). In the

experiments three different volumetric discharge as 11.02, 9.80, and 8.57 m³/hr based on the capacity of the circulation pump (Max-Min) are also used to conduct the experiment.

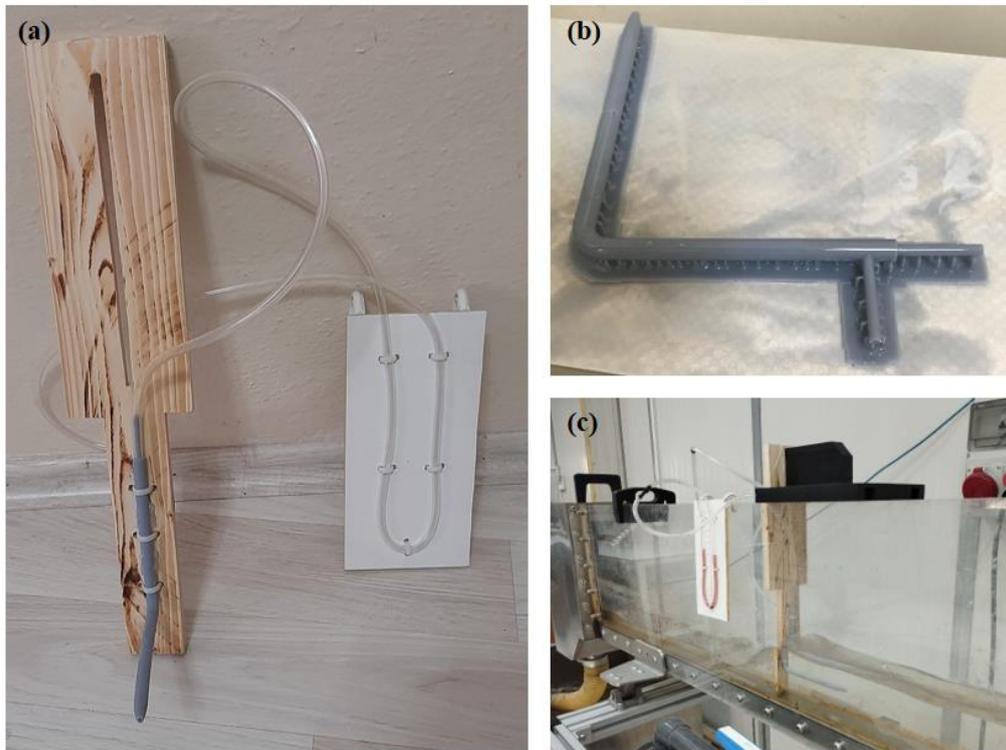


Figure 4. The ready to use (a) Pitot tube and the attached manometer, (b) the 3D printed Pitot tube, and (c) the conducted experiments in the open channel

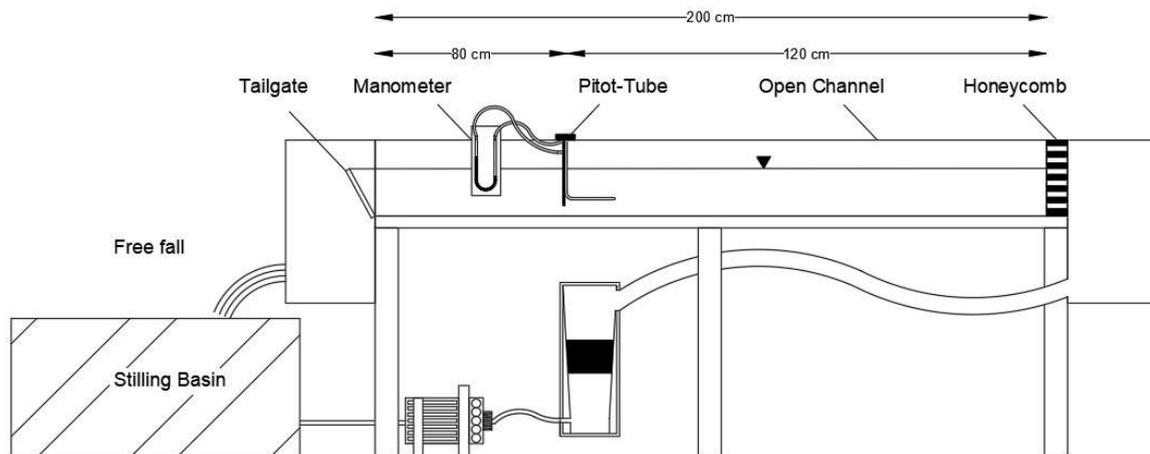


Figure 5. Open channel and location of the Pitot-tube

In the experiments, the pressure difference between points B and C in Fig. 6 is equal to the height (h) read on the manometer. Afterward, the Bernoulli's equation is applied by ignoring the height difference between two pipes as,

$$\frac{P_B}{\gamma} + 0 = \frac{P_C}{\gamma} + \frac{V^2}{2g} \quad (3)$$

Hence, the velocity (V) could be determined using,

$$V^2 = 2g \left(\frac{P_B - P_C}{\gamma} \right) = 2g \left(\frac{\Delta P}{\gamma} \right) \quad (4)$$

or

$$V = \sqrt{2gh} \quad (5)$$

while, P_B and P_C denotes the amount of pressure generated in the Pitot-tube tip and the holes, respectively. In addition, V represents the intended flow velocity, while γ , g , and h represent the specific weight of the water, the gravitational acceleration and head difference reads from the manometer, respectively. For this, h , the water level difference between the two pipes, is substituted in Eq. 5 to obtain the measured velocity of the flow (V_m).

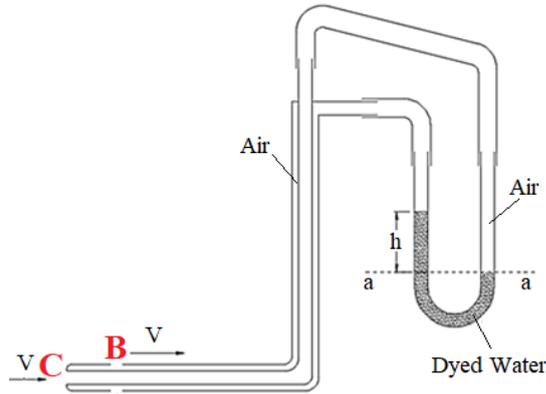


Figure 6. Combined Tube Manometer

Alternatively, the observed velocity (V_o) is determined using continuity equation as,

$$V_o = \frac{Q}{b \times h_c} \quad (6)$$

where Q is the flow rate read (i.e. 11.02, 9.80, and 8.57 m³/hr) by the flowmeter in the open channel; h_c is the water level in the channel and b is the channel width. The obtained V_o data is then used to evaluate the accuracy of the V_m values obtained by the Pitot-tube during the experiments. Therefore, the error (ε) is calculated as,

$$\varepsilon = \left| \frac{V_o - V_m}{V_o} \right| \times 100 \quad (7)$$

that could be used in evaluating the performance of the design.

3. Results

After which the Pitot-tube is fabricated, experiments are conducted in the open channel flow (at the 120 cm from the upstream) using 11.02, 9.80, and 8.57 m³/hr flow rates. In this stage, by considering the head difference (h_{dif}) between dynamic (h_D) and static (h_s) head in the pitot-tube the velocity of the current in the channel (Eq. 5) is obtained (V_m). Alternatively, the velocity of the current at the same point was calculated (V_o) with the help of flow depth, discharge and Eq. 6. Table 1, provides the obtained values for the aforementioned values, whilst the data are then used to determine the bias between V_o and V_m reads approximately as ε (1): %52. This determines that the relationship between V_o and flow rates is stronger than the V_m calculated with the Pitot-tube. Fig. 7a depicts the relationship of discharge rate (m³/hr) with V_o and V_m . The provided scatter diagram showed that the relationship is quite linear with high determination coefficient. But, the V_o rates are the twice of V_m rate (Fig. 7b). However, the

V_m reads could simply be transformed linearly to V_o with a help of a linear equation given in Fig. 7c. Hence, with a 92% concordance the provided linear equation can be introduced for further applications.

Table 1: Data evaluation chart (Q : discharge; h_D : dynamic head; h_s : static head; h_{dif} : head difference between h_D and h_s ; V_o : velocity of current (Eq.6); V_m : velocity of current (Eq.5); ε (1): the bias between V_o and V_m ; V_T : transferred velocity with the help of Equation in Fig 7c.; ε (2): the bias between V_o and V_T)

Exp. No.	Q (m ³ /hr)	h_D (cm)	h_s (cm)	h_{dif} (m)	V_o (m/s)	V_m (m/s)	ε (1) (%)	V_T (m/s)	ε (2) (%)
1	11.02	8.01	8.80	0.008	0.85	0.40	53.42	0.83	2.35
2	9.80	8.05	8.81	0.008	0.76	0.38	49.26	0.79	3.94
3	8.57	8.20	8.71	0.005	0.66	0.31	52.65	0.64	3.03
Mean	14.69	8.08	8.77	0.007	0.75	0.36	51.77	0.75	3.11

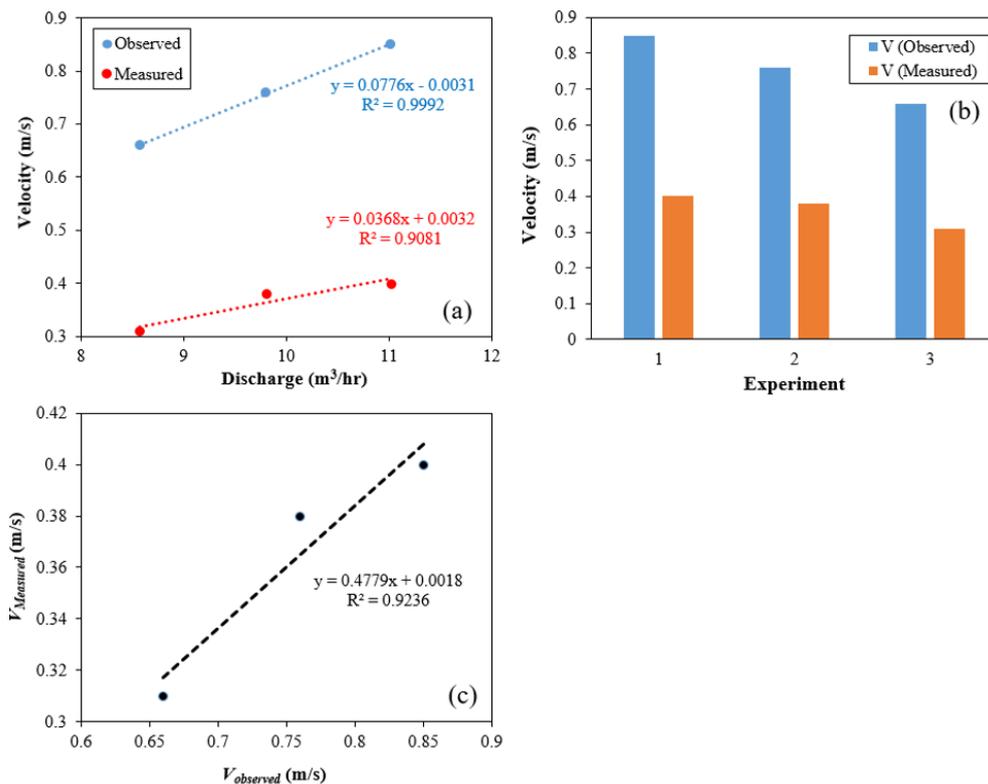


Figure 7. Results given as (a) a scatter plot of discharges vs. velocities and (b) bar chart of the V_o and V_m , and (c) the scatter plot of the observed vs. measured velocities

Therefore, the provided equation in Fig. 7c can be arranged as

$$V_T = \frac{V_m - 0.0018}{0.4779} \quad (8)$$

to obtain the transformed velocity reads (V_T) and use them instead of V_m in the comparison against V_o .

As also provided in Table 1, the mean values of the V_T and V_o matches perfectly, and the rate of bias sufficiently drops to ε (2): %3.11. Therefore, the fabricated equipment could be used with a help of an adjusting equation (i.e. Eq. 8). However, as the reads in the channel were conducted with the flowmeter and the classical volume-time method, the illustrated performance of the tube could be caused by the dimensions of the Pitot-tube which was preferred based on the sensitive commercial samples that are usually used in the measurement of the velocity of gases rather than liquids. When liquid is used, sometimes the capillary force in the tiny run between nested pipes cause blockage and avoids current flow. The geometry of the Pitot tube that can cause precision in the experiments however, was preferred with a pointy end which may cause in aerodynamic behavior, yet other samples with flat

forefront may present more accurate results. It is also known that the velocity distribution of the fluid through the cross section varies relative to the friction force originated from the interaction of the fluid with the solid surfaces (i.e. bottom and sidewalls) or the presence of turbulence volatilities. Our results, in concordance with the previous studies [21, 22], showed that the application of Pitot-tube in the open channel experiments is effective but also tricky. Yet, based on the preliminary studies conducted by Henry Darcy in 1856 and the latter review by the Brown [24] detailed experiments should be carried out to come across accurate reads. The limitation of this study, mainly is based on using alternative geometries in fabrication of the Pitot-tube, conducting relatively large number of experiments, and also using different scenarios (e.g. different discharge rates, bed slopes, cross sections, and depths).

4. Conclusion

In this study, the design principles of a Pitot-tube are addressed to obtain the velocity of water in an open channel of the laboratory-scaled studies. First, a Pitot tube is designed based on the principals provided in the statement of the art and then fabricated with the help of a 3D printer and liquid resin. In the next step, the fabricated Pitot-tube is connected to a Differential Manometer and used in the experiments. Observation rates are obtained by using the continuity equation, while the experimental velocity is determined with the help of the Manometer. Afterward, the observed and measured velocities are compared to determine the accuracy of the Pitot-tube reads. According to the initial results, the velocity of the Pitot tube has a relative 52% error compared to the classical reads. It has been concluded that different geometries and sensitivity analyses could possibly enhance further design attempts. But, with a help of linear equation, data reads are transferred and the relative error of 3% is achieved that could be considered as satisfactory.

Acknowledgements

This study was carried out within the scope of the BSc. Elective Graduation Project of the first author under the supervision of the second and third authors.

Nomenclature

The following abbreviations and symbols are used in the manuscript, while the symbols with different indices are detailed in the text.

Abbreviations		Symbols	
3D	3 Dimensional	γ	Specific weight [$M L^{-2} T^{-2}$]
ADV	Acoustic Doppler Velocity Meter	ε	Error (bias or precision)
FDM	Fused Deposition Modeling	b	Width of the channel [L]
PIV	Particle Image Velocimetry	g	Gravitational acceleration (9.81 m.s^{-2}) [LT^{-2}]
NRMSE	Normalized Root Mean Square Error	h	Height difference in differential Manometer [L]
SLA	Stereolithography	D	Outer diameter of the Pitot-tube [L]
UV	Ultra Violet	P	Pressure [$M L^{-1} T^{-2}$]
		Q	Discharge rate [L^3T^{-1}]
		R^2	Determination coefficient
		V	Velocity [LT^{-1}]

References

- [1] Cheng, N. S. (2007). Power-Law Index for Velocity Profiles in Open Channel Flows. *Advances in Water Resources*, 30(8), 1775-1784.
- [2] Węcel, D., Chmielniak, T., & Kotowicz, J. (2008). Experimental and numerical investigations of the averaging Pitot tube and analysis of installation effects on the flow coefficient. *Flow Measurement and Instrumentation*, 19(5), 301-306.
- [3] González, J. A., Melching, C. S., & Oberg, K. A. (1999). Analysis of open-channel velocity measurements collected with an acoustic Doppler current profiler. *1st International Conference on New/Emerging Concepts for Rivers*, IWR Association. Chicago, USA, September.
- [4] Hyun, B. S., Balachandar, R., Yu, K., & Patel, V. C. (2003). Assessment of Piv To Measure Mean Velocity And Turbulence in Open-Channel Flow. *Experiments in Fluids*, 35(3), 262-267.
- [5] Araújo, J. C. D., & Chaudhry, F. H. (1998). Experimental Evaluation of 2-D Entropy Model For Open-Channel Flow. *Journal of Hydraulic Engineering*, 124(10), 1064-1067.
- [6] Bagherimiyab, F., & Lemmin, U. (2013). Shear Velocity Estimates in Rough-Bed Open-Channel Flow. *Earth Surface Processes and Landforms*, 38(14), 1714-1724.
- [7] Chiu, C. L., & Said, C. A. A. (1995). Maximum and Mean Velocities And Entropy in Open-Channel Flow. *Journal of Hydraulic Engineering*, 121(1), 26-35.
- [8] Tarım Ve Orman Bakanlığı (2021). Hidrometrik Ölçüm Yönergesi. Ankara: Etüt, Planlama Ve Tahsisler.
- [9] Yurtseven, B.R., & Kalkavan, S. (2013). Hidrometri (Su ölçümü) Uygulamalarında Debi Ölçümleri. *DSİ*, 2020.
- [10] Meral, R., & Benli, A. (2013). Türkiye akarsuları akım ölçümlerinde mevcut durum ve alternatif yöntemlerin değerlendirilmesi. *Sakarya University Journal of Science*, 17(3), 477-481.
- [11] Klopfenstein Jr, R. (1998). Air velocity and flow measurement using a Pitot tube. *ISA transactions*, 37(4), 257-263.
- [12] Brown, G. O. (2003). Henry Darcy's perfection of the Pitot tube. Henry PG Darcy and other pioneers in hydraulics: contributions in celebration of the 200th birthday of Henry Philibert Gaspard Darcy. *American Society of Civil Engineers*, Reston, VA, 14-23.
- [13] Taylor, G. I. (1938). Measurements with a half-pitot tube. Proceedings of the Royal Society of London. Series A. *Mathematical and Physical Sciences*, 166(927), 476-481.
- [14] Korkmaz, S., & Ghaznawi, N. A. (2019). Numerical and Experimental Modeling of Flow Over a Broad-Crested Weir. *Sakarya University Journal of Sciences* 23(2), 252-258.
- [15] Care, I., & Fourneaux, F. (2020). Investigation of the pressure response of different Pitot tubes. *Flow Measurement and Instrumentation*, 72, 101714.
- [16] Ghaznawi, N. A. (2019). Geniş Tepeli Savak Üzerinden Geçen Akımın Sayısal ve Deneysel Modeli. Yüksek Lisans Tezi . Bursa: T.C. Bursa Uludağ Üniversitesi Fen Bilimleri Enstitüsü.
- [17] Demirel, İ. H. (2021). Küçük Akarsu Havzalarında Hidroenerji. Doktora Tezi. T.C. Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü.

- [18] Folsom, R. G. (1956). Review of the Pitot tube. *Transactions of the American Society of Mechanical Engineers*, 78(7), 1447-1460.
- [19] Berkün, M. (2019). Akışkanlar Mekaniği Ve Hidrolik. İstanbul: Literatür Yayınları.
- [20] Al-Hadhrami, L. M., Shaahid, S. M., Tunde, L. O., & Al-Sarkhi, A. (2014). Experimental study on the flow regimes and pressure gradients of air-oil-water three-phase flow in horizontal pipes. *The Scientific World Journal*.
- [21] Ghosh, S. N., & Roy, N. (1970). Boundary shear distribution in open channel flow. *Journal of the Hydraulics Division*, 96(4), 967-994.
- [22] Almeida, A. S., & Souza, V. C. B. D. (2017). An alternative method for measuring velocities in open-channel flows: performance evaluation of a Pitot tube compared to an acoustic meter (Technical Note). *Brazilian Journal of Water Resources*, 22(26), 1-14.
- [23] Brown, G. O. (2003). Henry Darcy's perfection of the Pitot tube. *In Henry PG Darcy and Other Pioneers in Hydraulics* (pp. 14-23). ASCE Library, USA.

A Simultaneous Numerical Integration Routine for the Fast Calculation of Similar Integrations

Aytac ALPARSLAN ^{1*} 

^{1*} Department of Electrical and Electronics Engineering, Faculty of Engineering, Trakya University, Edirne, 22030

Cite this paper as:

Alparslan, A. (2023). *A Simultaneous Numerical Integration Routine for the Fast Calculation of Similar Integrations*. Journal of Innovative Science and Engineering,7(2):133-141

*Corresponding author: Aytac Alparslan
E-mail: aytacalparslan@trakya.edu.tr

Received Date:02/02/2023
Accepted Date:21/08/2023
© Copyright 2023 by
Bursa Technical University. Available
online at <http://jise.btu.edu.tr/>



The works published in Journal of Innovative Science and Engineering (JISE) are licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Abstract

In this paper, a fast and simultaneous integration routine tailored for obtaining results of multiple numerical integrations is introduced. In the routine, the same nodes are used when integrating different functions along the same integration path. In the paper it is demonstrated by several examples that if the integrands of interest are similar on the integration path, then using the same nodes decreases the computational costs dramatically. While the method is introduced by updating the popular Gauss-Kronrod quadrature rule, the same steps given in the paper can be applied to any other numerical integration rule.

Keywords: Electromagnetic Line Sources, Gauss-Kronrod Quadrature, Layered Media Green's Functions, Numerical Integration, Simultaneous Integration.

1. Introduction

Numerical integration routines are indispensable tools in modern engineering. They are used in nearly every type of advanced engineering problems to obtain the integration result of a given integrand, especially when there is no closed-form analytical result [1–6]. In some problem types, the integration results of multiple integrands may be needed simultaneously. For instance, when obtaining the power flow generated by an electromagnetic source, all of the components of electric and magnetic fields are needed [7–9]. These quantities are usually obtained by computational electromagnetic methods such as finite element method [10], method of moments [11,12] or method of multipoles [13] where numerical integration routines are employed. The integrands of such integrations are usually similar in shape because the locations of the singularities, branch cuts and branch points are observed at the same locations on the integration plane. As a result of this fact, the distribution of nodes obtained by the adaptive numerical integration methods becomes very similar for all the integrands, if not the same. Using this observation, one can use the same set of nodes for all the integrations simultaneously, which eliminates the necessity of calculating the same functions when building up the integrands.

In this paper, a simultaneous integration routine that can be used to numerically integrate similar integrands is introduced. If the multiple integrands have same types of functions as their constituents and have the same distribution of singularities, branch points and cuts around the integration path, the introduced method decreases the computation costs of integrations dramatically.

The paper is organized as follows. In section 2, the method is described in detail by reminding the important aspects of the popular Gauss-Kronrod quadrature rule and the upgrade introduced in the paper. In section 3, numerical examples are given to demonstrate the efficiency and advantages of the method. In Section 4, the outcomes of the paper are listed.

2. Simultaneous Numerical Integration of Similar Integrands by Gauss-Kronrod Quadrature

One of the most popular numerical integration methods that is frequently used in computational electromagnetics is the Gauss-Kronrod rule, which is an upgraded version of the well-known n -point Gauss quadrature rule defined as follows [14]:

$$\int_a^b f(x)dx \approx G_n = \sum_{i=1}^n w_i f(x_i) = \vec{w} f(\vec{x}) \quad (1)$$

where w_i and x_i are the weights and nodes of the n -point Gauss quadrature rule, which are listed in the row and column vectors $\vec{w}_{(1 \times n)}$ and $\vec{x}_{(n \times 1)}$, respectively. In (1), $f(\vec{x}_{(n \times 1)})$ contains the integrand values evaluated at the nodes and it is a column vector with the same size as $\vec{x}_{(n \times 1)}$. This rule gives the exact integration results when the integrand $f(x)$ is at most an $(2n - 1)$ th degree polynomial, and therefore it is called a $(2n - 1)$ -degree quadrature rule. The n -point Gauss quadrature is one of the most efficient numerical integration schemes and it is widely used in the numerical analysis of different engineering problems. However, it cannot provide an error estimate of the numerical integration of arbitrary integrands in a single run, which can introduce problems when highly accurate results with fast computation times are needed. A solution to this problem was proposed by Alexander Kronrod [15]. He introduced $(n + 1)$ additional nodes (y_i) to the n -point Gauss quadrature rule to obtain the $(2n + 1)$ -point Gauss-Kronrod rule as follows:

$$\int_a^b f(x)dx \approx K_{2n+1} = \sum_{i=1}^n \alpha_i f(x_i) + \sum_{j=1}^{n+1} \beta_j f(y_j) = \vec{\alpha}f(\vec{x}) + \vec{\beta}f(\vec{y}) \quad (2)$$

A comparison between (1) and (2) reveals that both integration rules use the same n nodes at $\vec{x}_{(n \times 1)}$. When constructing the Gauss-Kronrod rule in (2) the total number of $(3n + 2)$ unknowns introduced by the new weight and node vectors $\vec{\alpha}_{(1 \times n)}$ ($\vec{\alpha}_{(1 \times n)}$ is not same as $\vec{w}_{(1 \times n)}$), $\vec{\beta}_{(1 \times (n+1))}$ and $\vec{y}_{((n+1) \times 1)}$ values makes it possible to obtain a $(3n + 1)$ -order quadrature rule [16]. In addition, a relative error estimate E of this rule can be readily obtained in a single run by comparing the result with n -point Gauss quadrature rule as follows:

$$E = \frac{|K_{2n+1} - G_n|}{|K_{2n+1}|} = \frac{|(\vec{\alpha} - \vec{w})f(\vec{x}) + \vec{\beta}f(\vec{y})|}{|\vec{\alpha}f(\vec{x}) + \vec{\beta}f(\vec{y})|} \quad (3)$$

In modern scientific computation software, such as Matlab [17,18], Octave [19] and quadpy [20], similar error estimates are used to build an adaptive Gauss-Kronrod integration scheme [21]. If an error criterion given by the user $E < \tau$ is not satisfied in an interval $[a, b]$, the Gauss-Kronrod rule is repeated on the sub-intervals $[a, (a + b)/2]$ and $[(a + b)/2, b]$. This division of intervals continues up until the error criteria is satisfied in all the sub-intervals. As a result, a robust numerical integration scheme is obtained that can be used efficiently in a variety of different engineering problems.

In some problem types, numerical integrations of similar integrands are needed. For example, when calculating the electromagnetic fields generated by a line source in planarly layered media by the Fourier type integrations, the same reflection and transmission coefficients are used to build up the spectral domain Green's functions. Since the results of the integrals, i.e., the spatial domain Green's functions, have the same type of plane wave constituents, the singularities, branch points and branch cuts are observed at the same locations in the spectral domain Green's functions defined for a given observation point. As a result, it is expected to have a similar, if not the same, distribution of nodes when performing the separate adaptive numerical integration of the Fourier type integrals by the Gauss-Kronrod quadrature rule as described above. By using this observation, the numerical integrations of different field components can be calculated simultaneously by using the same nodes (the same $\vec{x}_{(n \times 1)}$ and $\vec{y}_{((n+1) \times 1)}$ nodes in (2) are taken for different field components) along the same integration path. Consequently, the integrand is defined to be a row vector $\vec{f}_{(1 \times m)}(x)$ which contains the m similar integrand values at a given node x . If the node values are defined in a column vector as done in (1)-(3) $\vec{f}_{(1 \times m)}(\vec{x}_{(n \times 1)})$ becomes a matrix $\mathbf{F}_{(n \times m)}^x$, and likewise, $\vec{f}_{(1 \times m)}(\vec{y}_{((n+1) \times 1)})$ becomes a matrix $\mathbf{F}_{((n+1) \times m)}^y$. Therefore, the Gauss-Kronrod integration rule for the similar integrands can be written as follows:

$$\int_a^b \vec{f}(x)dx \approx \vec{K}_{2n+1} = \sum_{i=1}^n \alpha_i \vec{f}(x_i) + \sum_{j=1}^{n+1} \beta_j \vec{f}(y_j) = \vec{\alpha}\mathbf{F}^x + \vec{\beta}\mathbf{F}^y \quad (3)$$

In (4), \vec{K}_{2n+1} is a $(1 \times m)$ row vector that contains the results of the $(2n + 1)$ Gauss-Kronrod quadrature applied on the m similar integrands, $\vec{f}_{(1 \times m)}(x)$. As in the case of the single integrands, the adaptive integration scheme is applied as described until the error criteria is satisfied on all the sub-intervals, for all the integrands. It should be noted that for some integrands, this method will introduce more sub-intervals than necessary, but this will not affect the efficiency of the integration results and even lower numerical error values will be achieved for such integrands. Yet, an analysis would be needed to decide whether a simultaneous integration is needed or not depending on the constituents of the integrands. As discussed above, the calculation process of the fields generated by a line source in layered media is a case where the simultaneous numerical integration of the different field components would be used to obtain the results in a faster way compared to individual integrations. When using the simultaneous integration routine, the necessity of calculating the reflection and transmission coefficients separately for different field components is eliminated and the resulting numerical integrations are obtained within the same error criteria. The efficiency and the advantages of the method are demonstrated by numerical examples in the next section, both by analyzing general contour integrals with closed form results and the calculation performance of spatial domain Green's functions in layered media.

3. Numerical Examples

In this section, the numerical integration scheme introduced in the previous section is validated by examples that are frequently seen in engineering applications. The performance of the new simultaneous integration algorithm is compared with the individual integration algorithm.

3.1. Calculation of Contour Integrals with Analytical Results

In the first example, the integrals given in (5) with closed form analytical results found by the residue theorem [22] are numerically evaluated along the closed contour C shown in Figure 1. In (5), J_0 is the 0th order Bessel function of the first kind and $i = \sqrt{-1}$ is the unit complex number. In order to compare the performance of the different numerical integration routines analyzed in Chapter 2, the integrands are numerically integrated, first individually by the 15-point Gauss-Kronrod rule in (2) and then by the 15-point simultaneous integration rule derived in this paper in (4). For both integrations, the relative error stopping criteria are set to 10^{-10} and the initial number of nodes is set to 11, as it is done in most of the popular scientific software.

$$\begin{aligned}
 I_1 &= \oint_C \frac{J_0(2z)e^{i10z} - \cos(4z)}{(z - (0.5 - 0.1i))} dz \\
 I_2 &= \oint_C \frac{e^{i10z} - J_0(2z) + 2 \cos(4z)}{(2z - (1 - 0.5i))(z - (0.5 - 0.1i))} dz \\
 I_3 &= \oint_C \frac{e^{i10z} - 3J_0(2z) + 2 \cos(4z)}{(2z - (1 - 0.5i))(z - 0.33)(z - (0.5 - 0.1i))} dz \\
 I_4 &= \oint_C \frac{e^{i10z} + J_0(2z) \cos(4z)}{(z - 0.33)} dz
 \end{aligned} \tag{5}$$

$$I_5 = \oint_C \frac{e^{i10z} 0.5J_0(2z) + \cos(4z)}{(2z - (1 - 0.5i))} dz$$

$$I_6 = \oint_C \frac{e^{i10z} + J_0(2z) + \cos(4z)}{(z - 0.33)(z - (0.5 - 0.1i))} dz$$

$$I_7 = \oint_C \frac{J_0(2z)e^{i10z} + \cos(4z)}{(2z + (1 - 0.5i))(z + (0.5 - 0.1i))} dz$$

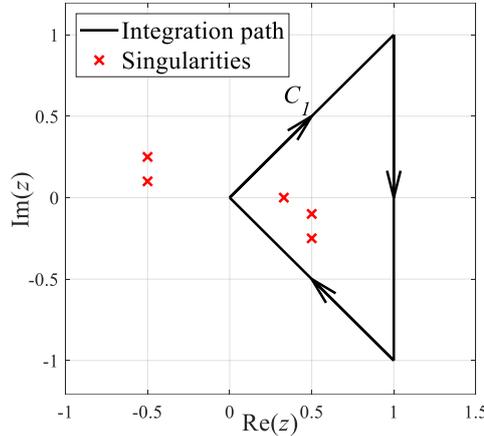


Figure 1. Integration path C used in (5) and the locations of the singularities of the integrands.

The results obtained by a single core of Intel(R) Xeon(R) CPU E5-2630 v3 using Matlab are shown in Table 1. As can be seen in the table, the relative errors compared with the analytical results are in the range of the stopping criteria for both integrations. The important difference is in the computation times, where an individual integration by (2) takes roughly 0.5ms and the simultaneous integration of all the integrands takes roughly 0.8ms. As a result, the total time for integrating all the 7 integrands is around 3.5ms which is around $\times 4.5$ slower than the simultaneous integration routine in (4). The main reason for this difference is eliminating the necessity of calculating the complex functions in the integrands repeatedly. When using the simultaneous integration routine, the functions are evaluated only once and used in all the integrands, since the same nodes are used in all the integrations. Whereas in the individual integration routine, a new set of nodes are used for all the different integrands.

Table 1. Numerical integration errors and computation times of the integrals in (5) obtained by the 15-point individual (2) and simultaneous (4) Gauss-Kronrod methods.

Integrand	Analytical result	Numerical Integration Error	Computation time (s)
I_1 (individual)	$- 1.455734953472314e + 01$ $- 8.014053318596627e + 00i$	$1.602939e - 13$ (relative)	$5.298821e - 04$
I_2 (individual)	$+ 4.744611107978165e + 01$ $- 1.638584608339527e + 02i$	$6.943940e - 15$ (relative)	$5.425719e - 04$
I_3 (individual)	$+ 7.754555018569379e + 00$ $- 1.126589747446449e + 02i$	$4.100069e - 15$ (relative)	$5.540496e - 04$
I_4 (individual)	$- 9.911454277117049e - 01$ $+ 4.810429771194812e + 00i$	$2.705469e - 13$ (relative)	$5.326485e - 04$
I_5 (individual)	$- 3.298687041869202e + 01$ $- 1.010583291349705e + 01i$	$2.700818e - 14$ (relative)	$5.437397e - 04$
I_6 (individual)	$- 3.940507818800279e + 01$ $- 5.785451053909134e + 01i$	$2.548616e - 14$ (relative)	$5.418304e - 04$
I_7 (individual)	$+ 0 + 0i$	$5.006176e - 13$ (absolute)	$5.494489e - 04$
All (simultaneous)	Same as listed above	$4.963638e - 13$ (max)	$7.569228e - 04$

3.2. Calculation of Spatial Domain Green's functions by Sommerfeld Integrations

In the second example, the calculation performance of spatial domain Green's functions defined for a line source placed in a multilayered geometry is analyzed by using different numerical integration methods. As discussed earlier, to obtain the different components of electric and magnetic fields generated by a line source, separate Fourier type numerical integrations are needed. Yet, the singularities, branch points and cuts stay at the same locations on the integration plane for a given layered medium, which makes such problems good candidates for using the simultaneous integration routine introduced in this paper. By using the new simultaneous integration routine, it is expected to obtain the results of the Fourier-type integrals faster than the individual integrations without losing the precision of the results. In order to validate this expectation, the Green's functions of a 3-layered medium composed of left-handed (LHM), and right-handed (RHM) materials are calculated, first individually by the 15-point Gauss-Kronrod rule in (2) and then by the proposed 15-point simultaneous integration rule in (4). The specifications of the problem is as follows: the free-space wavelength of the source is 500nm, the layers are stacked from bottom to top and have the following electromagnetic properties: layer-1 (RHM, silver) $\epsilon_{r1} = -9.621 + 0.31022i$, $\mu_{r1} = 1$, layer-2 (RHM, lossless) $\epsilon_{r2} = 2$, $\mu_{r2} = 1$, and layer-3 (LHM, lossy) $\epsilon_{r3} = -1 + 0.01i$, $\mu_{r3} = -1$. The thickness of the second layer is 500nm and layer 1 and 3 extend to infinity in y-direction. The magnetic-type line source is placed in the middle of layer-2. The magnitude of the Poynting vector \vec{S} generated by the line source defined by:

$$\vec{S} = \vec{E} \times \vec{H}^*, \quad (6)$$

can be seen in Figure 2, for the 0 longitudinal dependence wave vector k_y in (a) and $k_y = (0.1 + 0.1i)k_0$ in (b), where k_0 is the wavenumber in free space. Since both the individual and simultaneous plots are identical, only the results of the simultaneous integration are plotted in the figure. The detailed derivations of the related integrands and integration paths are provided in [23–26].

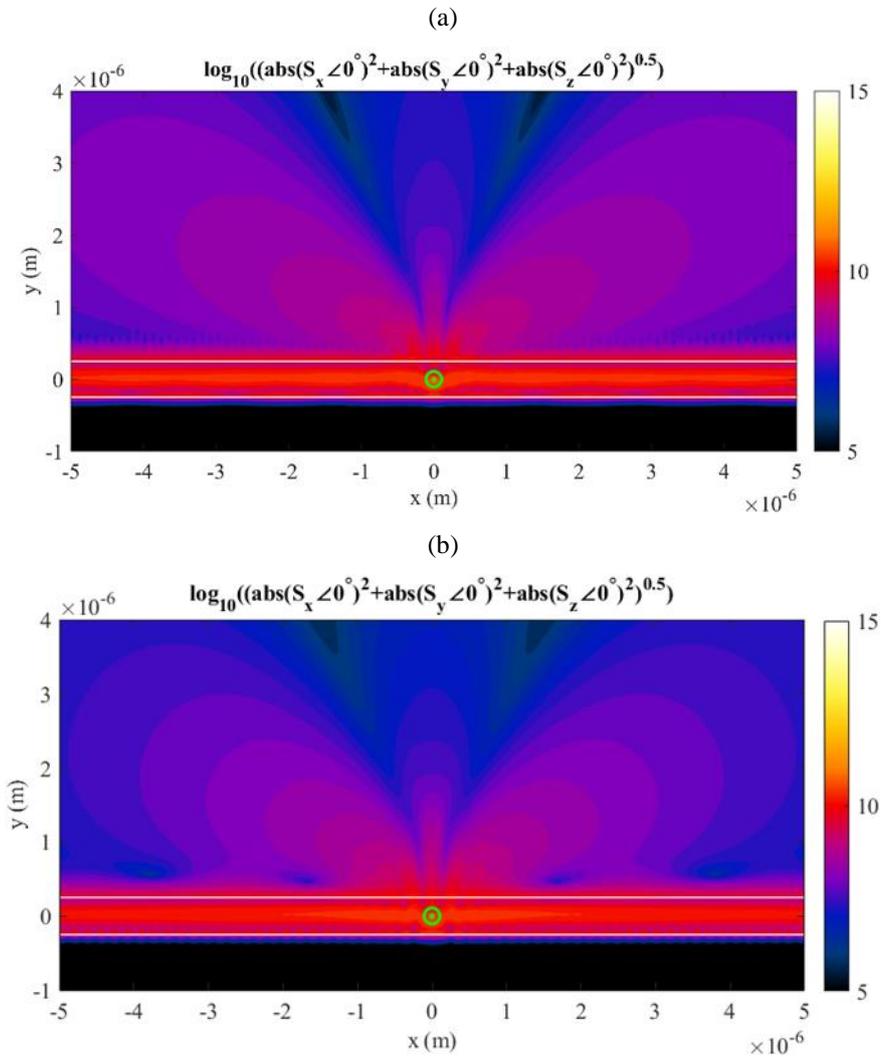


Figure 2. Magnitude of the Poynting vector in log scale generated by the magnetic line sources placed at the origin with the longitudinal dependence of (a): $k_y = 0$ and (b): $k_y = (0.1 + 0.1i)k_0$. Wavelength of the source is 500nm, layer-1 (bottom, RHM, silver) $\epsilon_{r1} = -9.621 + 0.31022i$, $\mu_{r1} = 1$, layer-2 (RHM, lossless) $\epsilon_{r2} = 2$, $\mu_{r2} = 1$, and layer-3 (LHM, lossy) $\epsilon_{r3} = -1 + 0.01i$, $\mu_{r3} = -1$. The thickness of the second layer is 500nm.

As discussed earlier, the integrands derived for the electric and magnetic fields have similar shapes on the integration path and taking the simultaneous numerical integrals decreases the computation times significantly, which are compared in Table 2. The results in Figure 2 and Table 2 are obtained by 16 cores of Intel(R) Xeon(R) CPU E5-2630 v3 using Matlab. The field values are calculated on the grid defined by the 400 linearly spaced points in x-direction between $-5 \mu\text{m}$ and $5 \mu\text{m}$ and the 200 linearly spaced points between $-1 \mu\text{m}$ and $4 \mu\text{m}$. Namely, the fields are calculated on the 80000 points on the field plane shown in Figure 2. The relative error stopping criteria of 10^{-6} is used in all the numerical integrations.

Table 2. Relative numerical integration errors and computation times of the Sommerfeld integrals when calculating the layered media Green's functions defined for the fields by the 15-point individual (2) and simultaneous (4) Gauss-Kronrod integrations. The calculated fields are used to plot Figure 2 by (6).

	Simultaneous ($k_\gamma = 0$)	Individual ($k_\gamma = 0$)	Simultaneous ($k_\gamma = (0.1 + 0.1i)k_0$)	Individual ($k_\gamma = (0.1 + 0.1i)k_0$)
Av. Error, E_x	$3.3622e - 09$	$2.0417e - 09$	$2.0417e - 09$	$2.3216e - 09$
Av. Error, E_y	$2.6046e - 09$	$1.9406e - 09$	$1.9406e - 09$	$1.9406e - 09$
Av. Error, E_z	N/A	N/A	$2.2708e - 12$	$1.1314e - 15$
Av. Error, H_x	N/A	N/A	$3.5034e - 09$	$3.5045e - 09$
Av. Error, H_y	N/A	N/A	$1.7555e - 07$	$1.8675e - 07$
Av. Error, H_z	$3.6108e - 09$	$3.6102e - 09$	$1.6110e - 09$	$1.6133e - 09$
Total time (s)	747	2135	1180	6817

Analyzing Table 2 reveals that, both integration methods give the results within the error criteria for the $k_\gamma = 0$ and $k_\gamma = (0.1 + 0.1i)k_0$ cases. The difference is observed in the calculation times, where the results are obtained nearly $\times 3$ times faster by the simultaneous integration routine when $k_\gamma = 0$ and nearly $\times 6$ faster when $k_\gamma = (0.1 + 0.1i)k_0$. This is the expected results as the magnetic line source generates 3 field components (E_x, E_y and H_z) when $k_\gamma = 0$ and it generates all the 6 field components when $k_\gamma \neq 0$. As a result of this comparison, it is shown that the simultaneous integration routine introduced in this paper can be used to obtain layered media Green's functions by the Sommerfeld integrations in a significantly faster way, without losing the accuracy of the results.

4. Conclusion

In this paper, a simultaneous numerical integration method is introduced. The main advantage of the proposed method is observed when a set of multiple integrations with similar integrands are needed. By using the method, the necessity of calculating the same functions repeatedly is eliminated and the integrands are obtained in a faster way. As a result, the integrals are calculated in a significantly faster way without decreasing the accuracy of the results. Examples are included to demonstrate the efficiency of the method by comparing the numerical integrations with analytical results and by analyzing the performance of the method when used in an advanced engineering problem.

References

- [1] Palacio-Betancur, A., & Gutierrez Soto, M. (2023). Recent Advances in Computational Methodologies for Real-Time Hybrid Simulation of Engineering Structures. *Archives of Computational Methods in Engineering*, 30(3), 1637–1662. <https://doi.org/10.1007/s11831-022-09848-y>
- [2] Jiang, M., Li, Y., Lei, L., & Hu, J. (2022). A Review on Fast Direct Methods of Surface Integral Equations for Analysis of Electromagnetic Scattering from 3-D PEC Objects. *Electronics*, 11(22), Article 22. <https://doi.org/10.3390/electronics11223753>
- [3] Botha, M. M. (2015). Numerical Integration Scheme for the Near-Singular Green Function Gradient on General Triangles. *IEEE Transactions on Antennas and Propagation*, 63(10), 4435–4445. <https://doi.org/10.1109/TAP.2015.2456959>
- [4] Davis, P. J., & Rabinowitz, P. (2007). *Methods of Numerical Integration*. Courier Corporation.

- [5] Kreyszig, E. (1999). *Advanced engineering mathematics* (8th ed.). J. Wiley.
- [6] Aimi, A., Diligenti, M., & Monegato, G. (1997). New Numerical Integration Schemes for Applications of Galerkin Bem to 2-D Problems. *International Journal for Numerical Methods in Engineering*, 40(11), 1977–1999. [https://doi.org/10.1002/\(SICI\)1097-0207\(19970615\)40:11<1977::AID-NME150>3.0.CO;2-O](https://doi.org/10.1002/(SICI)1097-0207(19970615)40:11<1977::AID-NME150>3.0.CO;2-O)
- [7] Felsen, N., & Marcuvitz, L. B. (1994). *Radiation and scattering of waves* (p. 924). John Wiley and Sons.
- [8] Chew, W. C. (1990). *Waves and fields in inhomogeneous media*. New York: Van Nostrand Reinhold.
- [9] Cheng, D. K. (1989). *Field and wave electromagnetics* (2nd ed.). Addison Wesley.
- [10] Jin, J.-M. (2015). *The Finite Element Method in Electromagnetics*. John Wiley & Sons.
- [11] Gibson, W. C. (2021). *The Method of Moments in Electromagnetics*. CRC Press.
- [12] Harrington, R. F. (1982). *Field Computation by Moment Methods*. Florida: Robert E. Krieger Publishing Company.
- [13] Hafner, C. (1999). *Post-Modern Electromagnetics: Using Intelligent Maxwell Solvers*. John Wiley and Sons.
- [14] Kahaner, D., Moler, C., & Nash, S. (1989). *Numerical methods and software*. Prentice-Hall, Inc.
- [15] Kronrod, A. S. (Aleksandr S. (1965). *Nodes and weights of quadrature formulas: Sixteen-place tables*. New York, Consultants Bureau. <http://archive.org/details/nodesweightsofqu0000unse>
- [16] Laurie, D. (1997). Calculation of Gauss-Kronrod quadrature rules. *Mathematics of Computation*, 66(219), 1133–1145. <https://doi.org/10.1090/S0025-5718-97-00861-2>
- [17] *MATLAB - MathWorks—MATLAB & Simulink*. (n.d.). Retrieved February 1, 2023, from <https://www.mathworks.com/products/matlab.html>
- [18] Shampine, L. F. (2008). Vectorized adaptive quadrature in MATLAB. *Journal of Computational and Applied Mathematics*, 211(2), 131–140. <https://doi.org/10.1016/j.cam.2006.11.021>
- [19] *GNU Octave*. (n.d.). Retrieved February 1, 2023, from <https://octave.org/index>
- [20] Schlömer, N. (n.d.). *quadpy: Numerical integration, quadrature for various domains* (0.16.22) [Python; OS Independent]. Retrieved February 1, 2023, from <https://github.com/sigma-py/quadpy>
- [21] Gonnet, P. (2012). A review of error estimation in adaptive quadrature. *ACM Computing Surveys*, 44(4), 22:1-22:36. <https://doi.org/10.1145/2333112.2333117>
- [22] Boas, M. (1983). *Mathematical methods in the physical sciences*. Wiley.
- [23] Alparslan, A. (2023). Constituents of electromagnetic 2-D layered media Green's functions for all material types and radiation conditions. *Waves in Random and Complex Media*, 1–26. <https://doi.org/10.1080/17455030.2023.2203264>
- [24] Michalski, K. A., & Mosig, J. R. (2015). The Sommerfeld half-space problem redux: Alternative field representations, role of Zenneck and surface plasmon waves. *IEEE Transactions on Antennas and Propagation*, 63(12), 5777–5790. <https://doi.org/10.1109/tap.2015.2489680>
- [25] Alparslan, A. (2013). *Numerical analysis of photonic nano structures in layered geometries* [ETH / ETH Zurich]. <https://doi.org/10.3929/ethz-a-010000418>
- [26] Aksun, M. I. (1996). A robust approach for the derivation of closed-form Green's functions. *Microwave Theory and Techniques, IEEE Transactions On*, 44(5), 651–658. <https://doi.org/10.1109/22.493917>

Arbitrary Phase Optimization Through Adaptively-Scheduled Nanophotonic Inverse Design

Emir Salih MAĞDEN¹ 

¹ Koç University, Department of Electrical & Electronics Engineering, 34450, İstanbul, Türkiye

Abstract

Design of integrated photonic devices continues to drive innovation in electro-optical systems for many applications ranging from communications to sensing and computing. Traditional design methods for integrated photonics involve using fundamental physical principles of guided-wave behavior to engineer optical functionalities for specific application requirements. While these traditional approaches may be sufficient for basic functionalities, the set of physically realizable optical capabilities by these methods remains limited. Instead, photonic design can be formulated as an inverse problem where the target device functionality is specified, and a numerical optimizer creates the device with appropriate geometrical features within specified constraints. However, even with inverse design methods, achieving arbitrarily-specified phase offsets on-chip remains an important problem to solve for the reliability of interferometry-based nanophotonic applications. In order to address difficulties in achieving simultaneous phase and power optimization in inverse nanophotonic design, in this paper, we develop a set of optimization approaches that can enable user-specified phase differences in single-wavelength and multi-wavelength nanophotonic devices. By specifying phase offset targets for each output, we prevent convergence failures resulting from the changes in the figure of merit and gradient throughout the iterative optimization process. Additionally, by introducing phase-dependent figure of merit terms through an adaptive scheduling approach during the optimization, we accelerate device convergence up to a factor of 4.4 times. Our results outline a clear path towards the optimization of nanophotonic components with arbitrary phase-handling capabilities, with potential applications in a wide variety of integrated photonic systems and platforms.

Keywords: Nanophotonic Design, Inverse Design, Silicon Photonics.

Cite this paper as:
Magden E.S. (2023). Arbitrary Phase Optimization Through Adaptively Scheduled Nanophotonic Inverse Design, 7(2):142-159

*Corresponding author: Emir Salih MAĞDEN
E-mail: esmagden@ku.edu.tr

Received Date: 18/08/2023.
Accepted Date: 05/10/2023.
© Copyright 2023 by
Bursa Technical University. Available
online at <http://jise.btu.edu.tr/>



The works published in the journal of Innovative Science and Engineering (JISE) are licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

1. Introduction

Integrated photonic devices and systems are one of the most important building blocks of systems used in communications, computing, and sensing applications today. With miniaturization of optical components and their integration on chip, these applications have seen tremendous growth in the last several decades [1]. Today, integrated photonic systems comprise some of the most critical components of these systems due to the advantages they provide in speed, efficiency, and multiplexing of information processing [2, 3]. Thanks to these advantages, typical building blocks components such as low-loss waveguides [4], broadband couplers [5], high-speed modulators [6], and low-noise detectors [7] have been previously demonstrated using fundamental understanding of guided-wave principles and coupled mode theory [8]. However, designing photonic devices for next-generation applications with complex, user-defined, and application-specific photonic functionality requires beyond what is possible with such traditional design thinking and fundamental physical principles [9, 10]. While these traditional photonic design principles rely on waveguide parameters and an understanding of optical coupling between them, the set of practically feasible photonic functions through these design principles remains limited [11].

In order to address the limitations of traditional photonic design, inverse design methods have recently emerged [12, 13]. Approaching the design problem from the reciprocal perspective, these approaches allow the user to define a specific objective, and then run an optimization process to create the device that can achieve this specified functionality with the appropriate geometrical features. Several examples have been demonstrated for building blocks including power splitters [14] as well as wavelength and mode multiplexers [15, 16]. Despite these demonstrations so far, arbitrary phase control of on-chip optical signals still remains an important problem to address in the general framework of inverse design approaches, as phase control is essential for interferometry-based on-chip communications and sensing. Particularly, it has been shown that while optical power at output waveguides of a multiplexer or splitter device can be easily optimized, achieving the same optimization performance for phase difference between the outputs can be prohibitively challenging [17]. A potential solution to this problem involves restricting the system design such that the outputs inherently achieve $\pi/2$ phase difference due to the underlying physical principles of mode interference [18]. Even though this approach may be sufficient for a restricted class of interferometric devices, achieving arbitrary phase differences between the optical outputs remains an important nanophotonic design problem to tackle.

In order to address this issue and build arbitrary phase handling capabilities on-chip, here we introduce an adaptively-scheduled phase optimization approach that can achieve user-specified phase differences between the output waveguides in both single-wavelength and multi-wavelength nanophotonic devices. Our approach relies on calculating the overlap of optical outputs with a reference profile, and adaptively introducing the amplitude and phase of this overlap into the objective function for the optimizer. Particularly, we demonstrate devices with arbitrarily-specified phase differences between multiple outputs, and illustrate the stability and faster convergence of this adaptive approach. We show that unlike directly including the phase difference of the outputs in the device objective, using a reference output mode profile ensures stability of the optimization process. We also demonstrate that our adaptive approach can achieve the same convergence in less than $1/4^{\text{th}}$ of the iterations required when phase information is directly included in the device performance calculation. These results illustrate the effectiveness of adaptive scheduling of objective function updates for on-chip design, and provide a clear path towards the

optimization of nanophotonic components with arbitrary phase-handling capabilities.

2. Methods

2.1. Device Geometry Representation Processing Operations

Optimization of a multi-output device to perform a pre-specified physical task under a given input can be cast as an inverse problem as illustrated in Figure 1. The problem is iteratively solved through forward and backward propagation steps. Starting with the forward propagation operation, we represent the nanophotonic device using a set of trainable parameters “ α ”, namely a density map of pixel values ranging between 0 and 1. We specify a physical pixel size of $25\text{nm} \times 25\text{nm}$, which also determines the spatial resolution of the discretized electromagnetic simulations we will run on these devices. This trainable map of pixels is held subject to a spatial filter with a normalized Gaussian kernel as $\alpha'(x, y) = \sum_{a, b \in D} \alpha(a, b) \kappa(a - x, b - y)$ where D is the region where the optimizable boundaries of the device are defined, and κ represents the spatial kernel. The radius of this kernel can be chosen depending on the restrictions of the fabrication platform used. Here we specify a kernel of radius of 200 nm (4 pixels) in order to sufficiently remove small geometrical features that are fabrication incompatible. A projection operation is then applied on this filtered permittivity to gradually minimize the number of pixels whose densities are far from the 0 and 1 boundaries. The goal of this projection is to ensure a final device with well-defined boundaries between the core and cladding materials chosen. This projection modifies the input α' as

$$\alpha_r = \frac{1}{2} + \frac{\tanh(\beta(\alpha' - 1/2))}{\tanh(\beta/2)} \quad (1)$$

where α_r is density map after projection operation, and β is parameter controlling the projection strength.

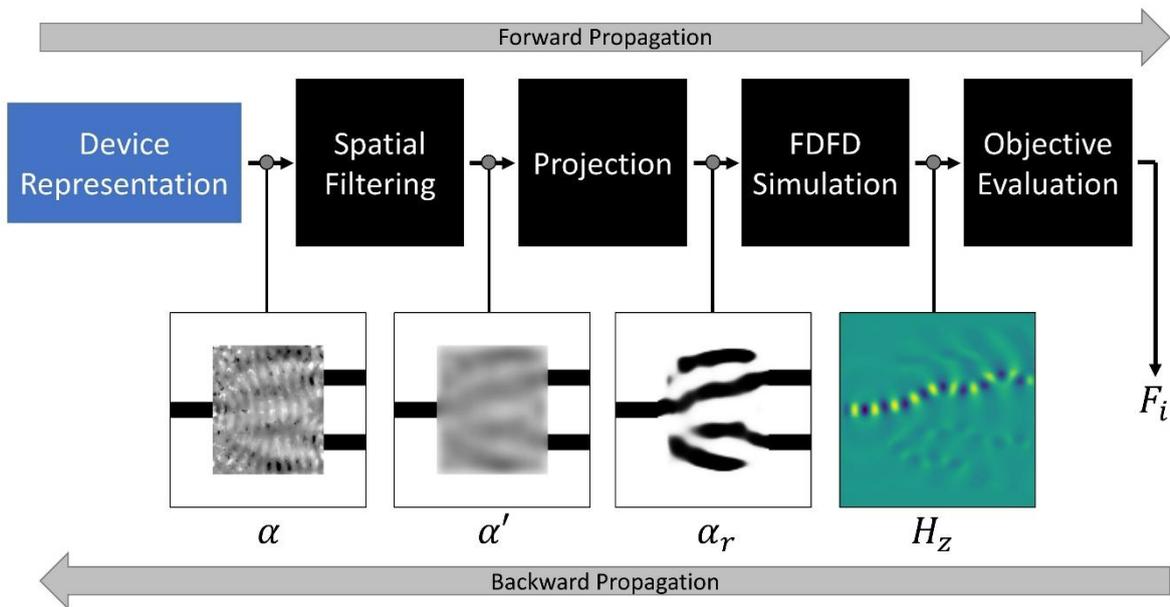


Figure 1. Device design procedure as an inverse problem with forward and backward propagation steps. Between each operation, the example output is shown by the two dimensional maps in the second row, and their corresponding mathematical symbols.

Using a larger β parameter increases the “discreteness” of the device, which we define as the ratio of pixels whose densities are in close proximity (within 2%) of 0 or 1. These spatial filtering radius and projection strength

parameters can be kept constant throughout optimization, or be modified gradually in order to ensure convergence and stability of the iterative process. In our demonstrations, we maintain a constant filter radius of 200 nm, but gradually increase the projection strength, as explained in the sections below.

Finally, the processed density map is linearly transformed into a distribution of relative permittivities. In this representation, each pixel is mapped to specific relative permittivity between ϵ_{Si} (Si “core” permittivity) and ϵ_{SiO_2} (SiO₂ “cladding” permittivity) of the silicon-on-insulator platform. For our devices, we consider 220nm-thick silicon-on-insulator platform as it is the most commonly used CMOS-compatible photonic platform currently. We also design them for operation at a wavelength of $\lambda = 1550\text{nm}$, due to the variety of existing applications and communication infrastructure at the C-band. The resulting relative permittivity for silicon is then calculated from the effective index of the fundamental transverse electric (TE) mode of a slab waveguide at a wavelength of 1550 nm as $\epsilon_{\text{Si}} = n_{\text{eff}}^2 = 2.84^2 = 8.07$. For the cladding, we directly use the refractive index of silicon dioxide at the same wavelength, yielding a relative permittivity of $\epsilon_{\text{SiO}_2} = n_{\text{SiO}_2}^2 = 2.07$.

Even though the design procedure shown in Figure 1 consists of many individual steps, the only parameter that physically makes up the device is the α_r , itself, which is a 2-dimensional map indicating the boundaries between the black (Si) and white (SiO₂) regions. Other parameters such as the spatial kernel (κ), the type or size of this kernel, the projection strength (β), or how/when this strength is updated are only used during the inverse design process. These parameters are not directly parts of the final device geometry; however, they serve an important intermediary purpose as they influence progression and convergence of the optimization process.

2.2. Photonic Device Performance Evaluation and Figures of Merit

The resulting geometrical structure’s electromagnetic response is simulated by using a finite difference frequency domain (FDFD) approach that solves

$$(-\omega^2 \epsilon_0 \epsilon_r(\mathbf{r}) + \mu_0^{-1} \nabla \times \nabla \times) \mathbf{E}(\mathbf{r}) = -i\omega \mathbf{J}(\mathbf{r}) \quad (2)$$

where ω is the frequency of the optical input, ϵ_0 is the permittivity of free space, $\epsilon_r(\mathbf{r})$ is the relative permittivity of the geometry after the operations described above, μ_0 is the permeability of free space, $\mathbf{E}(\mathbf{r})$ is the electric field distribution to be solved, and $\mathbf{J}(\mathbf{r})$ is the fundamental waveguide mode specified as the optical input to the simulation. This equation is then spatially discretized into a linear system in the form of

$$A \mathbf{E} = -i\omega \mathbf{J} \quad (3)$$

where A is the discretized representation of $(-\omega^2 \epsilon_0 \epsilon_r(\mathbf{r}) + \mu_0^{-1} \nabla \times \nabla \times)$, \mathbf{E} is the discretized solution of $\mathbf{E}(\mathbf{r})$, and \mathbf{J} is the discretized representation of the optical input $\mathbf{J}(\mathbf{r})$.

In this study, even though we focus only on two-dimensional simulations and TE optical inputs, our methods remain generalizable to three dimensions and also to the transverse magnetic (TM) polarization. For our TE simulations, using Eq. (2), we solve the non-zero components E_x , E_y , and H_z of the electric and magnetic fields using a parallel direct solver [19]. Following this solution, slices of these electric and magnetic fields in the y-direction are recorded as $H_{z\text{-out}(j)}(y)$, $E_{x\text{-out}(j)}(y)$, and $E_{y\text{-out}(j)}(y)$ on each output waveguide,. Then, the mode overlap, output phase, and the optical transmission in the x-direction are calculated for each output, using a reference slice $E_{\text{ref}}(y)$

extracted from the simulation of a straight waveguide using

$$\Gamma_{(j)} = \left| \int H_{z\text{-out}(j)}(y) E_{\text{ref}}(y) dy \right|^2 / \int |H_{z\text{-out}(j)}(y)|^2 dy \quad (4)$$

$$\theta_{(j)} = \arg \left(\int H_{z\text{-out}(j)}(y) E_{\text{ref}}(y) dy \right) \quad (5)$$

$$P_{x(j)} = -\frac{1}{2} \int \text{Re}\{E_{y\text{-out}(j)}(y) H_{z\text{-out}(j)}(y)\} dy \quad (6)$$

where (j) is the index of the output waveguide considered, $\Gamma_{(j)}$ is the mode overlap at the jth output waveguide, $\theta_{(j)}$ is the output phase at the jth output waveguide, and $P_{x(j)}$ is the x-directed transmission at the jth output waveguide. These performance metrics ($\Gamma_{(j)}$, $\theta_{(j)}$, and $S_{x(j)}$) describe the operation of the device for a given input. For most devices, while ideal operation means achieving a perfect mode overlap for all $\Gamma_{(j)}$, the relative phases between the outputs and the desired power transmission can vary depending on the application requirements. In order to handle arbitrary phase offsets and transmissions at the outputs, we first construct an overall figure of merit (FOM) as the following expression

$$F_1 = \sum_{j=1}^M (\Gamma_{(j)} - 1)^2 + \sum_{j=2}^M (\theta_{(j)} - \theta_{(1)})^2 + \sum_{j=1}^M (P_{x(j)} - T_{(j)})^2 \quad (7)$$

where F_1 is the typical figure of merit for multi-output device, M is the total number of outputs of the device, and $T_{(j)}$ is the desired target transmission at jth output waveguide. For most applications, the physical response of the device is influenced by the difference of the optical phases between each pair of outputs, and not directly by the value of $\theta_{(j)}$ itself. Therefore, the natural choice for the FOM shown above includes phase differences between output #1 and all the subsequent outputs. This means that the absolute phase of output #1 ($\theta_{(1)}$) is left as a free parameter during optimization. While this approach may seem suitable in general, as we will demonstrate in the following sections, significant changes in $\theta_{(1)}$ between iterations introduces instabilities in photonic design and optimization problems. To mitigate this behavior, we use the modified FOM given below.

$$F_2 = \sum_{j=1}^M (\Gamma_{(j)} - 1)^2 + \sum_{j=1}^M (\theta_{(j)} - \theta_{(j)}^{\text{target}})^2 + \sum_{j=1}^M (P_{x(j)} - T_{(j)})^2 \quad (8)$$

where F_2 is the modified figure of merit for multi-output device, and $\theta_{(j)}^{\text{target}}$ is the target phase at the jth output waveguide. In this modified FOM, we specify a target phase $\theta_{(j)}^{\text{target}}$ for all of the device outputs, leaving no free parameters. For the devices we demonstrate in this paper, we set $\theta_{(1)}^{\text{target}} = 0$ for simplicity; but this phase can be arbitrarily chosen as desired. The target phases at the other outputs may be chosen as necessary for each individual device.

We extend these expressions for use in multi-wavelength devices by adding wavelength-dependent terms for all performance metrics calculated. As before, we use two separate FOM constructions in order to illustrate the effects

of adding individual phase targets for each output. In this case, $\Gamma_{(j)}$, $\theta_{(j)}$, and $P_{x(j)}$ are all wavelength-dependent; and we have the two following FOM descriptions:

$$F_3 = \sum_{k=1}^K \left(\sum_{j=1}^M (\Gamma_{(j)}(\lambda_k) - 1)^2 + \sum_{j=2}^M (\theta_{(j)}(\lambda_k) - \theta_{(1)}(\lambda_k))^2 + \sum_{j=1}^M (P_{x(j)}(\lambda_k) - T_{(j)}(\lambda_k))^2 \right) \quad (9)$$

$$F_4 = \sum_{k=1}^K \left(\sum_{j=1}^M (\Gamma_{(j)}(\lambda_k) - 1)^2 + \sum_{j=1}^M (\theta_{(j)}(\lambda_k) - \theta_{(j)}^{\text{target}}(\lambda_k))^2 + \sum_{j=1}^M (P_{x(j)}(\lambda_k) - T_{(j)}(\lambda_k))^2 \right) \quad (10)$$

where F_3 is the typical figure of merit for multi-wavelength and multi-output device, F_4 is the modified figure of merit for multi-wavelength and multi-output device, K is the total number of wavelengths, and λ_k is the k^{th} wavelength in the summation. As before, the typical multi-wavelength FOM F_3 is constructed with phase differences calculated with respect to output #1, whereas the modified multi-wavelength FOM F_4 defines target phases $\theta_{(j)}^{\text{target}}(\lambda_k)$ for all outputs at each wavelength.

We use FDFD simulations in order to obtain $\Gamma_{(j)}$, $\theta_{(j)}$, and $P_{x(j)}$ parameters and the resulting figures of merit. Instead, some studies in literature use finite-difference time-domain (FDTD) simulations for similar optimizations [20, 21]. While FDTD is known for its physical accuracy, it also introduces significant computational bottlenecks in the optimization procedure due to the computationally lengthy and memory-intensive time-stepping calculations required for obtaining electric and magnetic fields [22, 23]. Reported optimizations with FDTD-based simulations can last as long as several days [24], which significantly limit the usability and repeatability of such optimization procedures for different applications. In contrast, FDFD-based methods like those that we use here are computationally simpler as they require no time-stepping, and can be generally solved using direct (non-iterative) matrix methods [25, 26]. These advantages allow us to design functionally complex and novel devices within reasonable timeframes (ranging from several minutes to about one hour), depending on the desired functionality. Perfectly matched layers (PML) were used as boundaries in all of our simulations for absorption of any light that is incident on the boundaries. Specifically, we used 20-layers (500 nm) thick PML regions on all sides of the simulation window to ensure sufficient light absorption and physical accuracy. The use of these PML boundaries also helps design low-loss devices, as the optimizer inherently minimizes any optical power lost to the PML boundaries while maximizing power transmission at the output waveguides.

2.3. Gradient-Based Optimization

Once the above objectives are specified for a particular device, the goal of the optimization procedure is to minimize the specified FOM quantity by iteratively modifying the device geometry. In order for this optimization to take

place, the optimization framework calculates the gradient of the chosen FOM with respect to the design parameters, through a backpropagation procedure as denoted in Figure 1. The efficient calculation of this gradient is enabled by the reverse-mode automatic differentiation capability in the open-source pytorch software library [27].

Specifically, all mathematical operations illustrated in Figure 1 including the field calculation in Eq. (3), the evaluation of performance metrics in Eq. (4)-(6), and the calculation of all FOM expressions in Eq. (7)-(10) are constructed using automatic differentiation-compatible functions. These functions are then automatically mapped to a cascading of primitive mathematical operations whose derivatives are known in pytorch. This allows the gradient of the FOM with respect to the design variables “ α ” to be efficiently and quickly calculated using chain rule through the specific cascade of operations used.

Using this gradient information, the FOM is minimized by modifying the design variables through a gradient-based descent algorithm. For the devices in the following sections, we use the limited-memory implementation of a bounded version of Broyden–Fletcher–Goldfarb–Shanno algorithm (L-BFGS-B) [28] to minimize the chosen FOM, as it has been shown to work well for photonic design problems in the past [26]. The basic principle of this optimization process relies on iteratively minimizing the calculated FOM by taking gradual steps towards the direction of an expected local minimum. The direction and size of this step is generally dictated by the calculated gradient of the FOM, as well as approximations of the FOM’s Hessian obtained from successive gradient calculations. In practice, even though L-BFGS-B can perform multiple FOM and gradient evaluations at each step, the resulting smoother progression yields a more reliable convergence to an acceptable local minimum than simple gradient descent, making it an appropriate choice for photonic design problems. For our devices, all of these optimization procedures were carried out on a workstation PC with an Intel Xeon processor using eight cores.

2.4. Updating of Projection Strength During Optimization

In prior studies, typical projection strengths on the order of $\beta = 100$ were used to make sure that optimizers yield sufficiently discrete and fabrication-compatible devices [25]. However, recent studies have shown that starting with a continuous structure (low β), and gradually increasing the projection strength to slowly approach a discrete structure (high β) improves optimization performance [21]. However, instead of using pre-determined increase intervals, we use an approach where the iterations at which projection strength is updated are not known a-priori. For our demonstrations, we initialize the projection strength at $\beta = 10$, and double it until a maximum of $\beta = 320$. The doubling occurs at iterations where the chosen FOM reaches below a threshold, typically specified between 5×10^{-3} and 1×10^{-2} .

2.5. Adaptively-Scheduled Introduction of Phase Terms in the Figures of Merit

In addition to using phase terms in the modified figures of merits F_2 and F_4 , we also use an adaptively-scheduled approach to start including these terms in the optimization process. First, we run optimizations with these figures of merit as expressed directly in their current form. Then we compare this with a novel, adaptive scheduling approach where the phase-dependent terms $\theta_{(j)}$ are introduced only after certain conditions are met in the device geometry. In this adaptive method, we calculate and include the phase information only after an FOM of approximately 5×10^{-3} to 8×10^{-3} is obtained and the device reaches a discreteness of at least 50-70%. We demonstrate in the following sections that the adaptive introduction of phase information into the FOM significantly

improves optimization performance.

3. Results and Discussion

3.1. Demonstration of a 1×3 Nanophotonic Splitter with Arbitrarily-Defined Output Phase Offset

We first illustrate the capability of our design procedure by designing a 1-input, 3-output power splitter that achieves specific phase differences between the outputs. The target transmissions are specified to be all equal at the three outputs as $T_{(1)} = T_{(2)} = T_{(3)} = 1/3$. We impose that the output phases such that $\theta_{(2)} - \theta_{(1)} = \pi/2$ and $\theta_{(3)} - \theta_{(1)} = \pi/3$ for our typical FOM F_1 , which correspond to $\theta_{(1)} = 0$, $\theta_{(2)} = \pi/2$, and $\theta_{(3)} = \pi/3$ for the modified FOM F_2 . These specific phase offsets are only used here as examples to demonstrate the capability of the design approach, and can be arbitrarily specified depending on the application requirements. The device is optimized within a relatively large footprint of $10\mu\text{m} \times 10\mu\text{m}$, due to the difficulty of achieving a phase-dependent device response compared to standard nanophotonic power splitter problems. The selection of these dimensions also determines the final footprint of the resulting device, as the optimizer is configured to only modify the density map of Si and SiO₂ within this specified region. The fundamental TE mode of a 500nm-wide waveguide is chosen as the optical input to the device. The optimization progress using the typical FOM F_1 is plotted in Figure 2(a), for four separate and randomly chosen initial conditions as shown by the red curves. In all four cases, the optimization experiences abrupt termination due to unsuccessful minimization operations in the direction the gradient, and ultimately the failure of convergence within the first few tens of iterations. The spikes observed in all FOM calculations correspond to updates of the projection strength β . However, due to optimization failure, the number of projection updates also remains insufficient to achieve a discrete and fabrication-compatible device. In contrast, the optimization performed with our modified FOM F_2 is plotted in in Figure 2(b). Here, the optimization procedure is successfully completed in 465 iterations as indicated by the orange curve. The final FOM of 10^{-3} is used as the stop condition of this optimization, at which point further improvements in performance metrics $\Gamma_{(j)}$, $\theta_{(j)}$, and $S_{x(j)}$ are observed to be negligible. The spikes at iterations 50, 152, 244, 281, and 308 bring the final projection strength to $\beta = 320$, resulting in a more reliable convergence to an acceptable minimum, and a fabrication-compatible final device. More importantly, the comparison between F_1 (red) and F_2 (orange) illustrates the success of our modified figure of merit calculation, as the optimization progress completes successfully, with no termination problems due to abrupt changes in FOM or its gradient. Compared to a pre-determined optimization progress, adaptive and real-time updating of these parameters helps achieve best device functionality while conforming to the specific nonlinear dynamics of the FOM surface as a function of device parameters.

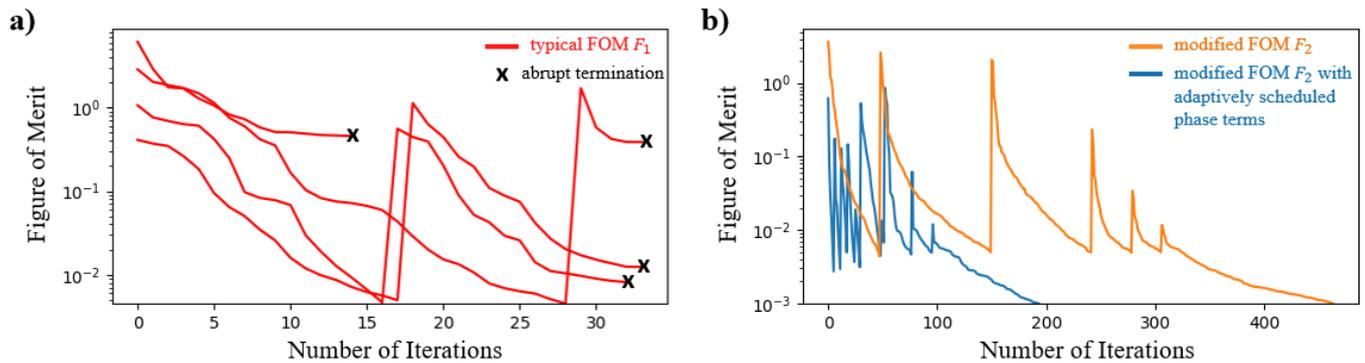


Figure 2. The optimization progress for a nanophotonic phase-dependent power splitter with one input and three outputs showing (a) typical FOM F_1 and (b) modified FOM F_2 . For the case of F_2 , standard (orange) and adaptively-scheduled (blue) approaches are plotted separately. In the adaptively-scheduled approach, the phase-related terms are introduced once the FOM is below 7×10^{-3} and the device satisfies at least 70% discreteness. Spikes in FOM correspond to increases in projection strength.

We also demonstrate that introducing the phase-dependent terms in FOM F_2 later on during the optimization significantly improves the number of iterations required for convergence. This is illustrated by the blue curve in Figure 2 for which only the first and third terms from Eq. (7) are considered initially. The second term with phase calculations is introduced in the FOM only after an FOM of 7×10^{-3} is obtained and the device reaches a discreteness of at least 70%. For this specific device, these conditions are met after iteration 27 of the optimization, once the mode overlap and transmission terms already achieve an acceptably low FOM on their own. As evident by the comparison of the orange and blue curves, this adaptively-scheduled phase optimization method achieves the same FOM much faster, in less than half the number of iterations. This can be attributed to the device already being located near a local minimum at the end of the first part of optimization without the phase terms, making the subsequent optimization of output phase offsets significantly easier, even after the FOM surface is modified as the new phase terms added.

The optimized device obtained at the end of this procedure and its FDFD-simulated H_z field are shown in Figure 3. Here, the final device geometry is shown with the spatial distribution of Si (black) and SiO₂ (white) regions, within the allocated optimizable area. As expected from an inverse design approach, the device geometry obtained from the optimizer is physically non-intuitive, and cannot be interpreted through fundamental guided wave principles [9, 17, 25]. Yet, the field shows near-perfect coupling of 1/3 power to the output ports, as indicated by the even field amplitudes at each output ports. Moreover, in contrast to previous optimizers that cannot enforce output phases [5, 14], this specific example also achieves the desired phase offset between output pairs. The smooth and large geometrical features of the final device indicate the qualitative success of the spatial filtering and projection operations used throughout the optimization.

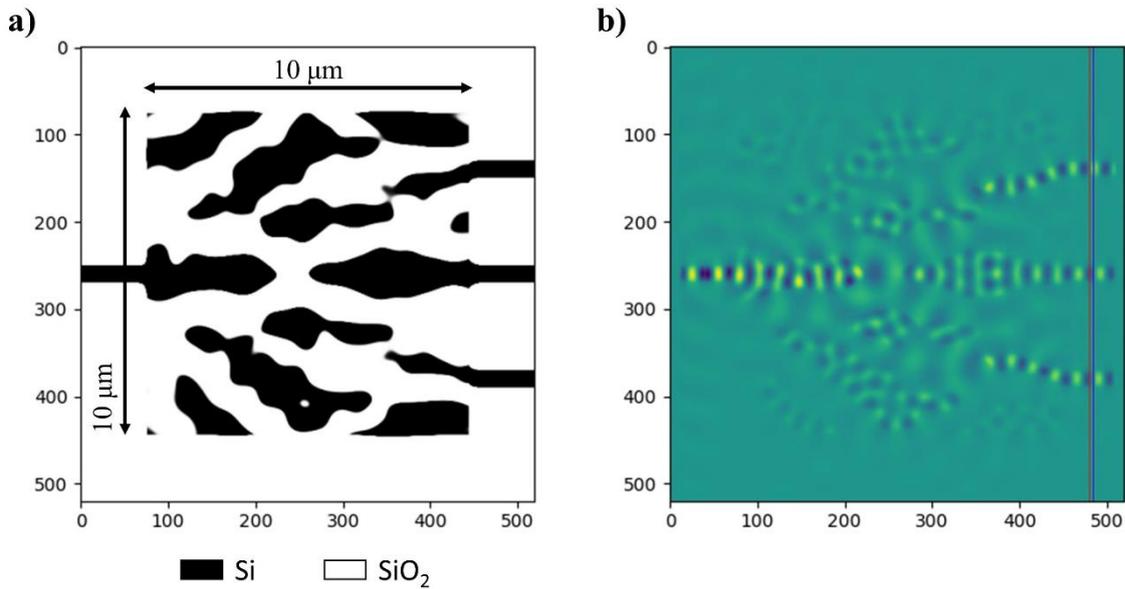


Figure 3. (a) Final device geometry for a 1-input and 3-output device, with a $10\mu\text{m} \times 10\mu\text{m}$ footprint, using an SOI platform with a 220 nm-thick silicon device layer. (b) The resulting H_z field achieving even power splitting with $\pi/2$ and $\pi/3$ phase offsets at the output waveguides. All axes indicate number of pixels where each pixel corresponds to physical dimensions of $25\text{nm} \times 25\text{nm}$. The red and blue vertical slices mark the output plane of the device where the phase information is recorded.

The phase offset between the specific outputs is better illustrated with the close-up of the FDFD result shown in Figure 4. In this figure, we crop the H_z result at the three output waveguides, and plot them together to demonstrate how the output phase of the fields relate to each other. During optimization, this phase difference is calculated as the angle of the complex overlap integral in Eq. (6), but can also be extracted using the longitudinal spacing between the consecutive wave maxima as shown. Between outputs 1 and 2, the positions of the output field maxima are separated by a distance of approximately $\Delta L_{21} = 144\text{nm}$. This corresponds to a phase difference of $\theta_{(2)} - \theta_{(1)} = (2\pi/\lambda)n_{\text{eff}}\Delta L_{21} = 0.53\pi$ where $n_{\text{eff}} = 2.84$ is the effective index of the TE slab mode as before. This matches well with the desired phase offset of $\pi/2$ that is used for the calculation of the FOM. Similarly, a phase difference of 0.36π is measured using the longitudinal position offset of $\Delta L_{31} = 97\text{nm}$ between the wave maxima at outputs one and three. This measurement once again matches well with the target phase difference of $\pi/3$ that was used in the FOM calculation. These results prove the effectiveness of the adaptively-scheduled FOM calculation in achieving arbitrarily-specified phase differences between the optical outputs.

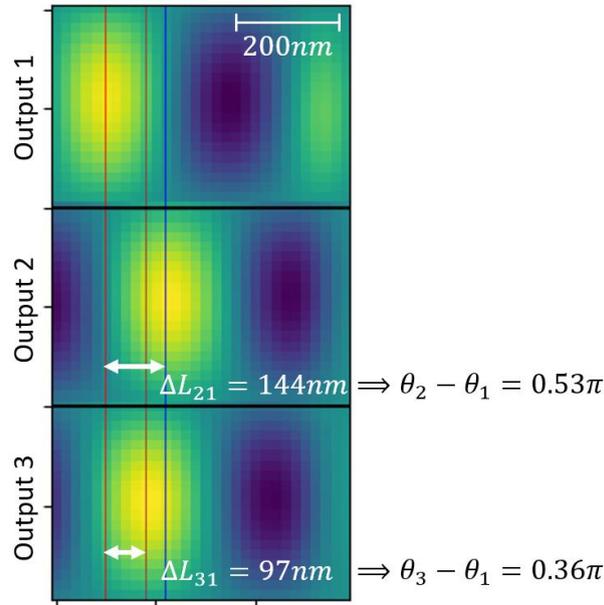


Figure 4. Close-up view of the output waveguides showing the guided wave output. The vertical lines indicate the positions of the wave maxima. The longitudinal separation of these maxima determine the phase difference between the outputs for each waveguide pair as shown.

The progression of these performance metrics throughout the adaptively-scheduled optimization procedure we described is shown in Figure 5. As with successful device optimizations, we observe the convergence of the phase offsets and the transmissions to the specified targets at the end of optimization. More specifically, during the first stage of optimization where no phase calculations are considered, the transmissions at each one of the three outputs tend gradually towards the specified target of $1/3$. The iteration at which phase terms are introduced in the optimization is marked with the red, vertical dashed line. The terms included in the FOM are indicated above the corresponding red arrows shown. As soon as the phase terms are included, the transmissions experience a significant drop due to the sudden modification of the FOM calculation. However, the optimizer quickly recovers and successfully modifies the device in order to achieve near lossless and even transmission at each output port. The final transmissions at each output are recorded as 0.317, 0.317, and 0.310. The overall insertion loss of the device is calculated as 0.25 dB. This result is consistent with the design of similar couplers in literature, even though these prior demonstrations do not include phase-dependent performance metrics [13, 14, 29]. As expected, we also observe the convergence of phase offsets $\theta_{(2)} - \theta_{(1)}$ and $\theta_{(3)} - \theta_{(1)}$ within close proximity of the desired $\pi/2$ and $\pi/3$ targets respectively, by the end of optimization.

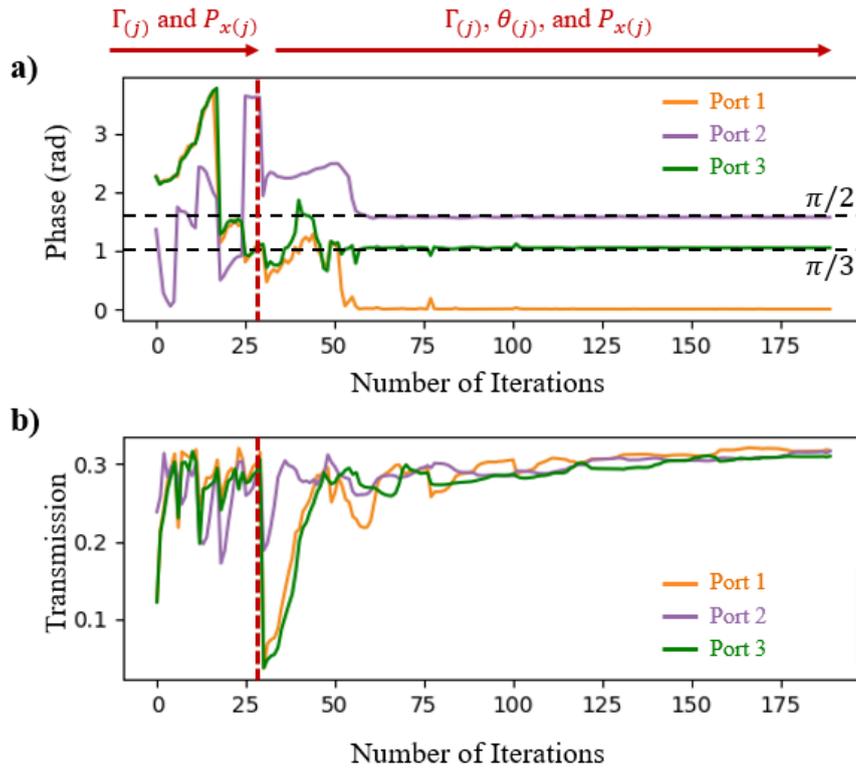


Figure 5. Evolution of (a) phase and (b) optical transmission at each output port of the 1-input 3-output device. Dashed lines indicate target phase offsets ($\theta_{(2)} - \theta_{(1)} = \pi/2$ and $\theta_{(3)} - \theta_{(1)} = \pi/3$) included in the FOM calculation.

3.2. Demonstration of a 1×2 Nanophotonic Splitter with Wavelength-Dependent Phase Offsets

Our design procedure with arbitrarily defined phase offsets between the outputs and their adaptive inclusion in the FOM calculation extends beyond single wavelength devices. When specifying the device objective, these phase offsets can also be configured as a function of wavelength as expressed by the FOMs F_3 and F_4 in Eq. (8) and (9). For this demonstration, we choose four operating wavelengths between 1500nm and 1600nm, and specify target phase offsets as a function of this wavelength as shown in Table 1. These offsets can be arbitrarily chosen as before, and are only used as examples for this specific device. As before, we use the fundamental TE mode of a 500nm-wide waveguide as the optical input to the device. The target transmission at each output port of the device is specified to be $1/2$ for even splitting, as required by many interferometry applications. As before, we also target a mode overlap of $\Gamma_{(j)} = 1$ at all outputs and for all wavelengths. For this device with a more complex and wavelength dependent FOM, we specify an optimizable design area of $12\mu\text{m} \times 12\mu\text{m}$, in order to provide sufficient degrees of freedom to the optimizer. Similar to the previous device, a spatial Gaussian kernel of 200 nm radius and a projection with gradually increasing strength is applied in order to ensure fabrication compatibility of the final structure.

Table 1. Wavelength-Dependent Phase Offset Specifications for Output Waveguides

Wavelength (nm)	Desired Phase Offset (radians)
1500	0
1530	$\pi/3$
1560	$2\pi/3$
1600	$\pi/2$

The device is optimized using these specifications with an L-BFGS-B optimizer as shown by the progression in

Figure 6. First, the typical FOM optimization procedure is performed using F_3 where the progress in the FOM is plotted by the red curves in Figure 6(a). Each curve represents a separate and randomly chosen initial condition for the device. As before, all examples face abnormal termination due to unpredictable changes in the FOM and gradient information, resulting in unsuccessful device optimizations. These devices also do not reach the stages with sufficiently large projection strengths necessary for fabrication-compatible results. In contrast, the modified approach where we include phase targets for all outputs using F_4 is shown by the orange curve in Figure 6(b). Including all terms in Eq. (10) in the FOM from the beginning of optimization, the iterative design procedure completes in 2001 iterations with a final FOM of 4×10^{-3} . After this threshold, the device performance exhibits only negligible changes. Due to gradual increases in the projection strength starting with $\beta = 10$, sudden increases in the FOM are observed at iterations 394, 629, 1381, 1539, and 1737 until $\beta = 320$ is reached. The increased number of iterations here compared to the previous 1×3 device can be attributed to the increased complexity of device functionality, for achieving wavelength-dependent phase offsets at the outputs. Once again, our modified approach with F_4 where phase targets are explicitly specified for each output demonstrates successful convergence; whereas F_3 with only the relative phase difference terms fails to converge.

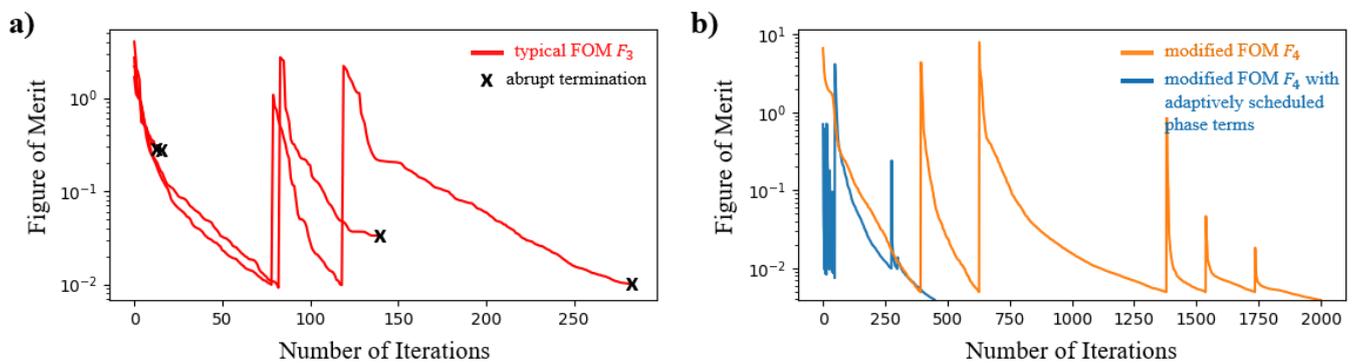


Figure 6. The optimization progress for a multi-wavelength, nanophotonic, phase-dependent power splitter with one input and two outputs showing (a) typical FOM F_3 and (b) modified FOM F_4 . For the case of F_4 , standard (orange) and adaptively-scheduled (blue) approaches are plotted separately.

We also implement our adaptive scheduling of phase-dependent terms in the FOM calculation for this multi-wavelength device. In this approach, the terms that depend on $\theta_{(j)}(\lambda_k)$ in Eq. (9) at all wavelengths are included only after a FOM of 8×10^{-3} is reached, and the device achieves a discreteness of above 50%. These conditions are met relatively quickly (only after 49 iterations) as indicated by the blue curve, as the device operates simply as a 3 dB splitter without the phase-dependent functionality. At iteration 50, when the phase-dependent terms are added in the FOM calculation, the plotted FOM experiences a sudden jump as the device no longer operates near a local minimum of this updated FOM. However, this difference is recovered relatively quickly; and the device achieves the same convergence criterion as the standard method in 449 iterations. As a result, our adaptive scheduling of phase terms improves convergence speed by a factor of over 4.4 times (in less than $1/4^{\text{th}}$ of iterations). The combination of separate phase target specification for all outputs and their adaptively-scheduled introduction into the FOM allow for successful and efficient nanophotonic device optimizations with simultaneous multi-wavelength, multi-output, and phase-dependent capabilities.

The final geometrical structure and the output fields for this device are plotted in Figure 7. Similarly, the regions corresponding to Si and SiO₂ in the final device are shown with black and white, respectively. Once again, the optimizer returns a physically non-intuitive, yet functionally near-perfect device at the end of the design process.

From a qualitative perspective, the transmission achieved at each output port for all four wavelengths is visually even, with only small amounts of field visibly lost towards the simulation boundaries. Moreover, despite the difficulty of the both wavelength-dependent and phase-dependent FOM specified, the final device exhibits relatively large and smooth geometrical features as well as a fully binary structure with only Si (black) and SiO₂ (white) regions, due to the spatial filtering and projection operations used in the design process. These qualities point to the overall success of the optimizer at designing low-loss devices with the desired characteristics.

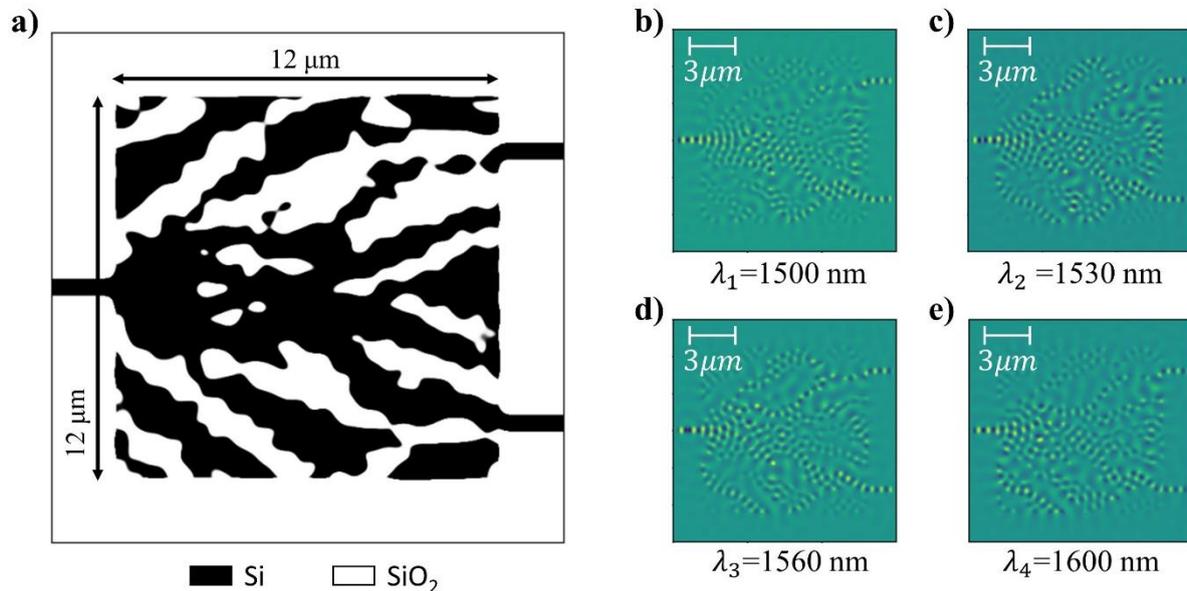


Figure 7. (a) Optimized 1 x 2 device with wavelength-dependent phase offsets at output ports, with a 12 μm × 12 μm footprint, using an SOI platform with a 220 nm-thick silicon device layer. (b)-(e) Spatial profiles of H_z at all four optimization wavelengths demonstrating even power splitting with different phase offsets. Scale bars added for reference.

More specifically, the effects of gradual updating of design parameters on the transmission and phase characteristics of the device are plotted in Figure 8 and Figure 9. Due to the increments in the projection strength β , the change in the output transmissions at the two ports undergo significant and sudden changes before iteration 49. After the phase terms are introduced, these transmissions gradually recover and settle around several percentage points below the 1/2 target as shown in Figure 8. Specifically, the device exhibits at most 0.2 dB insertion loss at these four design wavelengths, which is comparable to previously demonstrated devices even though no phase-dependent objectives were reported [13, 29, 30]. In contrast, the phase difference between the outputs is only optimized after iteration 49, as the phase-related terms are included in the updated FOM calculation. Once these terms are included, the phase difference between the two outputs quickly achieves the design targets within less than only 15 iterations, as demonstrated by the convergence behavior in Figure 9. This indicates the relative ease of achieving phase-dependent objectives in nanophotonic splitters, whereas simultaneously achieving desired transmission characteristics can be more difficult as shown by the gradual improvements in Figure 8 from iteration 49 until the end of optimization. As a result, our demonstrated findings present a straightforward and methodical approach towards achieving phase-dependent output profiles, unlike prior approaches that modify the input field as a function of the desired output phases [17].

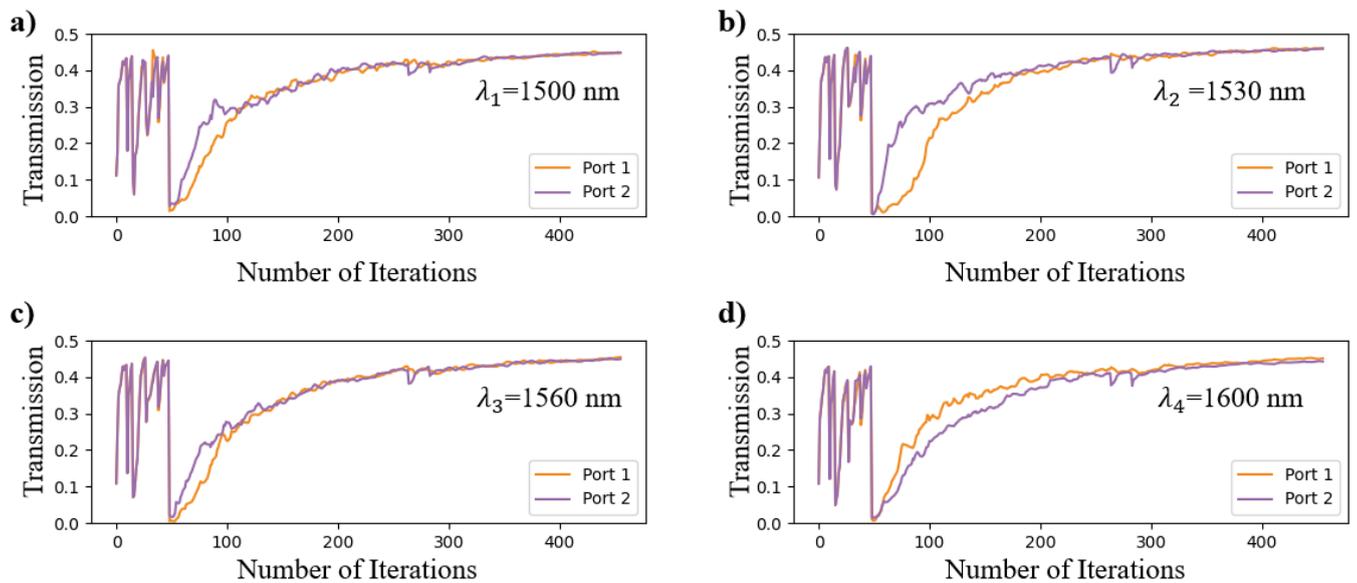


Figure 8. Progression of transmission at the two outputs of nanophotonic splitter with wavelength-dependent phase offsets at (a) 1500 nm, (b) 1530 nm, (c) 1560 nm, and (d) 1600 nm.

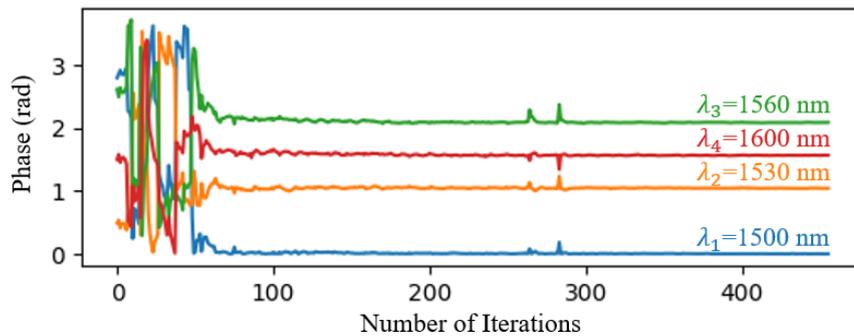


Figure 9. Progression of phase difference between the two outputs of the 1x2 nanophotonic splitter at four wavelengths throughout optimization.

3.3. Computational Performance of Adaptively-Scheduled Device Optimization

To the best of our knowledge, in addition to being one of the only optimization approaches that robustly and consistently enables arbitrary phase offsets at the outputs with remarkable device performance, our phase-scheduled algorithms also demonstrate superior computational efficiency. More importantly, our optimization approach does not introduce any additional or mathematically intensive calculations other than the simple phase extraction shown in Eq. (5), and therefore maintains the inherent computational efficiency of frequency-domain approaches. Specifically, for our first device (1×3 splitter), the entire optimization shown by the blue curve in Figure 2(b) completes in 280 s, resulting in an average of 1.4 s per each iteration. As already anticipated, compared to other FDTD-based methods that could result in optimizations lasting several days [14, 24, 31, 32], this result illustrates the ability of our demonstrated algorithm to reach convergence many orders of magnitude times faster. The asymptotic computational time and memory complexity of our FDFD-based algorithm is the same as other FDFD-based methods [26, 33], as the underlying operations for a system of Maxwell's equations solutions remain unchanged. As a result, our resulting per-iteration computational times are consistent with comparable frequency-domain optimizations in literature [25]. Yet, our optimization algorithm can reach overall convergence faster than

several of these previously demonstrated results, due to the adaptive introduction of selected terms in the figures of merit, consequently enabling convergence in fewer iterations.

In contrast, our second (1×2 splitter) device with wavelength-specific phase differences targets a significantly more challenging objective than a single-wavelength device. As a result, the optimization shown by the blue curve in Figure 6(b) takes significantly longer and completes in 3825 s (about 64 min), corresponding to 8.5 s per iteration (approximately 2.1 s per FDFD simulation at each one of the four wavelengths). These simulations at each wavelength are approximately 1.5 times slower than our first device, since this second device has a larger footprint ($12\mu\text{m} \times 12\mu\text{m}$ vs. $10\mu\text{m} \times 10\mu\text{m}$), and a greater number of resulting discretized cells. As such, even though the asymptotic computational complexity is the same as our previous device, the larger simulation area, simulations repeated at each wavelength, and the more complex target objective contribute to a longer overall optimization. Still, the final result is obtained much faster than the FDTD-based methods, and in consistent time scales as other FDFD-based methods referenced above.

4. Conclusion

In summary, we have demonstrated a novel and adaptive method for reliably and consistently achieving arbitrarily-specified phase differences between the outputs of nanophotonic splitters. Instead of only using phase differences, by including phase targets for all outputs in the FOM calculations, we have shown that iterative optimizers like L-BFGS-B can be configured to robustly and reliably design devices with specific phase offsets. Additionally, we have also demonstrated that instead of including these phase-dependent terms right at the beginning of optimization, adaptively scheduling their introduction later on during the optimization process drastically improves the overall convergence speed. Even though our adaptive scheduling method is one of the only approaches demonstrated so far that can reliably enable output phase offsets, this capability does not sacrifice the underlying computational efficiency of FDFD simulations. Overall computational times vary from several minutes to just over one hour for the two types of devices we demonstrated, in agreement with previous frequency-domain optimizers. Our demonstrated method works well for the design of single-wavelength and multi-wavelength devices as shown above, making it applicable for a large set of photonic systems in optical communications, sensing, and computing.

Acknowledgements

This study was supported by Scientific and Technological Research Council of Turkey (Türkiye Bilimsel ve Teknolojik Araştırma Kurumu - TÜBİTAK) under grant number 119E195.

References

- [1] Thomson, D., Zilkie, A., Bowers, J. E., Komljenovic, T., Reed, G. T., Vivien, L., Marris-Morini, D., Cassan, E., Virot, L., and Fédéli, J.-M. (2016). Roadmap on silicon photonics. *Journal of Optics*, 18(7), 073003.
- [2] Liu, A., Liao, L., Chetrit, Y., Basak, J., Nguyen, H., Rubin, D., and Paniccia, M. (2009). Wavelength division multiplexing based photonic integrated circuits on silicon-on-insulator platform. *IEEE Journal of Selected Topics in Quantum Electronics*, 16(1), 23-32.

- [3] Dong, P. (2016). Silicon photonic integrated circuits for wavelength-division multiplexing applications. *IEEE Journal of Selected Topics in Quantum Electronics*, 22(6), 370-378.
- [4] Heck, M. J., Bauters, J. F., Davenport, M. L., Spencer, D. T. and Bowers, J. E. (2014). Ultra-low loss waveguide platform and its integration with silicon photonics. *Laser & Photonics Reviews*, 8(5), 667-686.
- [5] Lu, Z., Yun, H., Wang, Y., Chen, Z., Zhang, F., Jaeger, N. A., and Chrostowski, L. (2015). Broadband silicon photonic directional coupler using asymmetric-waveguide based phase control. *Optics express*, 23(3), 3795-3808.
- [6] Witzens, J. (2018). High-speed silicon photonics modulators. *Proceedings of the IEEE*, 106(12), 2158-2182.
- [7] Piels, M. and Bowers, J. E. (2023). Photodetectors for silicon photonic integrated circuits. *Photodetectors*, 419-436.
- [8] Bogaerts, W. and Chrostowski, L. (2018). Silicon photonics circuit design: methods, tools and challenges. *Laser & Photonics Reviews*, 12(4), 1700237.
- [9] Molesky, S., Lin, Z., Piggott, A. Y., Jin, W., Vucković, J., and Rodriguez, A. W. (2018). Inverse design in nanophotonics. *Nature Photonics*, 12(11), 659-670.
- [10] So, S., Badloe, T., Noh, J., Bravo-Abad, J. and Rho, J. (2020). Deep learning enabled inverse design in nanophotonics. *Nanophotonics*, 9(5), 1041-1057.
- [11] Jensen, J. S. and Sigmund, O. (2011). Topology optimization for nano-photonics. *Laser & Photonics Reviews*, 5(2), 308-321.
- [12] Wiecha, P. R., Arbouet, A., Girard, C. and Muskens, O. L. (2021). Deep learning in nano-photonics: inverse design and beyond. *Photonics Research*, 9(5), B182-B200.
- [13] Piggott, A. Y., Petykiewicz, J., Su, L. and Vučković, J. (2017). Fabrication-constrained nanophotonic inverse design. *Scientific reports*, 7(1), 1786.
- [14] Tahersima, M. H., Kojima, K., Koike-Akino, T., Jha, D., Wang, B., Lin, C., and Parsons, K. (2019). Deep neural network inverse design of integrated photonic power splitters. *Scientific reports*, 9(1), 1368.
- [15] Jia, H., Zhou, T., Fu, X., Ding, J. and Yang, L. (2018). Inverse-design and demonstration of ultracompact silicon meta-structure mode exchange device. *ACS Photonics*, 5(5), 1833-1838.
- [16] Piggott, A. Y., Ma, E. Y., Su, L., Ahn, G. H., Sapra, N. V., Vercruyssen, D., Netherton, A. M., Khope, A. S., Bowers, J. E., and Vuckovic, J. (2020). Inverse-designed photonics for semiconductor foundries. *ACS Photonics*, 7(3), 569-575.
- [17] Hammond, A. M., Slaby, J. B., Probst, M. J. and Ralph, S. E. (2022). Phase-Injected Topology Optimization for Scalable and Interferometrically Robust Photonic Integrated Circuits. *ACS Photonics*, 10(4), 808-814.
- [18] Guan, H., Ma, Y., Shi, R., Zhu, X., Younce, R., Chen, Y., Roman, J., Ophir, N., Liu, Y., and Ding, R. (2017). Compact and low loss 90° optical hybrid on a silicon-on-insulator platform. *Optics Express*, 25(23), 28957-28968.

- [19] Schenk, O. and Gärtner, K. (2004). Solving unsymmetric sparse systems of linear equations with PARDISO. *Future Generation Computer Systems*, 20(3), 475-487.
- [20] Zeng, Z., Venuthurumilli, P. K. and Xu, X. (2021). Inverse design of plasmonic structures with FDTD. *ACS Photonics*, 8(5), 1489-1496.
- [21] Hammond, A. M., Oskooi, A., Johnson, S. G. and Ralph, S. E. (2021). Photonic topology optimization with semiconductor-foundry design-rule constraints. *Optics Express*, 29(15), 23916-23938.
- [22] Li, Z., Pestourie, R., Park, J.-S., Huang, Y.-W., Johnson, S. G., and Capasso, F. (2022). Inverse design enables large-scale high-performance meta-optics reshaping virtual reality. *Nature communications*, 13(1), 2409.
- [23] Hammond, A. M., Oskooi, A., Chen, M., Lin, Z., Johnson, S. G., and Ralph, S. E. (2022). High-performance hybrid time/frequency-domain topology optimization for large-scale photonics inverse design. *Optics Express*, 30(3), 4467-4491.
- [24] Piggott, A. Y., Lu, J., Lagoudakis, K. G., Petykiewicz, J., Babinec, T. M., and Vučković, J. (2015). Inverse design and demonstration of a compact and broadband on-chip wavelength demultiplexer. *Nature Photonics*, 9(6), 374-377.
- [25] Hughes, T. W., Minkov, M., Williamson, I. A. and Fan, S. (2018). Adjoint method and inverse design for nonlinear nanophotonic devices. *ACS Photonics*, 5(12), 4781-4787.
- [26] Minkov, M., Williamson, I. A., Andreani, L. C., Gerace, D., Lou, B., Song, A. Y., Hughes, T. W., and Fan, S. (2020). Inverse design of photonic crystals through automatic differentiation. *Acs Photonics*, 7(7), 1729-1741.
- [27] Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. (2017). Automatic differentiation in pytorch.
- [28] Yuan, Y.-x. (1991). A modified BFGS algorithm for unconstrained optimization. *IMA Journal of Numerical Analysis*, 11(3), 325-332.
- [29] Chang, W., Ren, X., Ao, Y., Lu, L., Cheng, M., Deng, L., Liu, D., and Zhang, M. (2018). Inverse design and demonstration of an ultracompact broadband dual-mode 3 dB power splitter. *Optics Express*, 26(18), 24135-24144.
- [30] Xu, J., Liu, Y., Guo, X., Song, Q. and Xu, K. (2022). Inverse design of a dual-mode 3-dB optical power splitter with a 445 nm bandwidth. *Optics Express*, 30(15), 26266-26274.
- [31] Li, R., Zhang, C., Xie, W., Gong, Y., Ding, F., Dai, H., Chen, Z., Yin, F., and Zhang, Z. (2023). Deep reinforcement learning empowers automated inverse design and optimization of photonic crystals for nanoscale laser cavities. *Nanophotonics*, 12(2), 319-334.
- [32] Xu, K., Liu, L., Wen, X., Sun, W., Zhang, N., Yi, N., Sun, S., Xiao, S., and Song, Q. (2017). Integrated photonic power divider with arbitrary power ratios. *Optics letters*, 42(4), 855-858.
- [33] Hughes, T. W., Williamson, I. A., Minkov, M. and Fan, S. (2019). Forward-mode differentiation of Maxwell's equations. *ACS Photonics*, 6(11), 3010-3016.

N-Boc-Amino Acid Mediated Morita-Baylis Hillman Reaction of Methylphenyl Glyoxylate

Gamze KOZ^{1*} , Necdet COSKUN² 

^{1*} Department of Chemistry, Faculty of Engineering and Natural Sciences, Bursa Technical University, 16310, Bursa, TURKEY

² Department of Chemistry, Faculty of Arts and Sciences, Bursa Uludag University, 16059, Bursa, TURKEY

Cite this paper as:

Koz, G. and Coskun, N. (2023). *N-Boc-Amino Acid Mediated Morita-Baylis Hillman Reaction of Methylphenyl Glyoxylate*. Journal of Innovative Science and Engineering, 7(2):160-166

*Corresponding author: Gamze Koz
E-mail: gamze.koz@btu.edu.tr

Received Date: 27/07/2023

Accepted Date: 16/10/2023

© Copyright 2023 by

Bursa Technical University. Available
online at <http://jise.btu.edu.tr/>



The works published in Journal of Innovative Science and Engineering (JISE) are licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

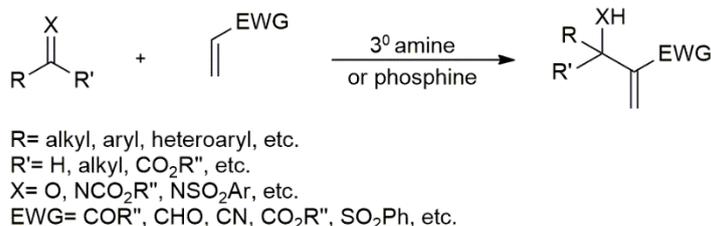
Abstract

The organocatalyzed Morita-Baylis Hillman (MBH) reaction of α -keto esters is a challenging carbon-carbon bond-forming reaction. We developed a catalytic system for the MBH reaction of methylphenyl glyoxylate with methyl vinyl ketone in a polar aprotic solvent. We used N-Boc-L-pipecolinic acid as a proton transfer mediator and 4-dimethylaminopyridine as the tertiary amine catalyst. We obtained the MBH adduct with a 66% yield in 48h. We proposed a detailed reaction mechanism involving a transition state that includes the hydrogen transfer by the acid functional group of N-Boc-L-pipecolinic acid.

Keywords: Morita-Baylis Hillman reaction, α -Keto ester, N-Boc-L-pipecolinic acid, 4-Dimethylaminopyridine.

1. Introduction

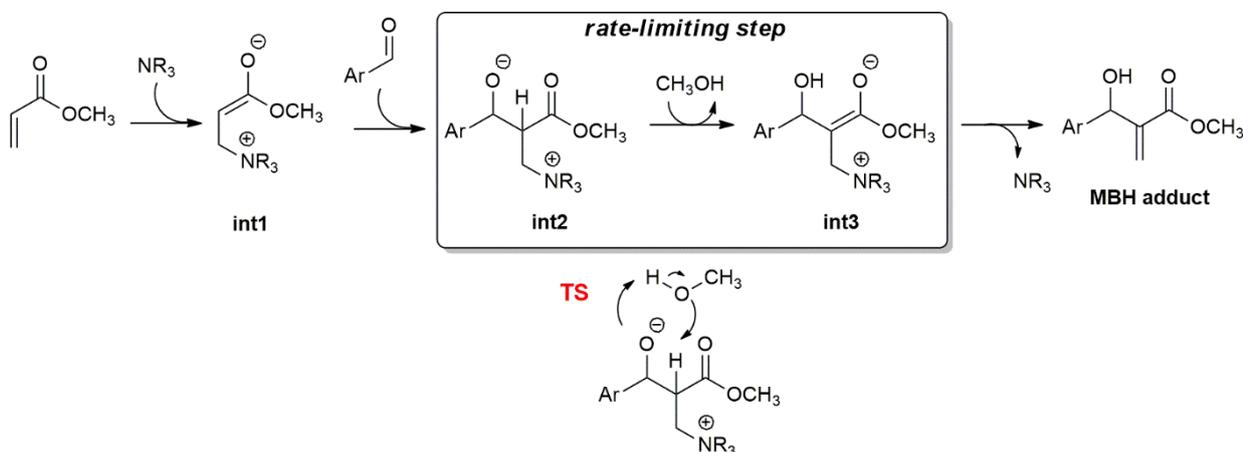
The carbon-carbon (C-C) bond-forming reactions are one of the most powerful tools in organic chemistry. Morita-Baylis Hillman (MBH) reaction is a practical C-C bond-forming reaction resulting in densely-functionalized products in the presence of Lewis bases such as tertiary amines and tri-substituted phosphines. Triphenylphosphine catalyzed traditional MBH reaction is between an aromatic aldehyde and methyl vinyl ketone [1] but can be extended to any activated alkene attached to a strong electron-withdrawing group and an electrophilic carbon such as an aldehyde or an imine (Scheme 1) [2-5].



Scheme 1. The classical MBH reaction catalyzed by amines and phosphines

MBH reaction has attracted more attention in recent years due to the commercially available starting materials, its atom-economical nature, multifunctional products and organocatalyzed mild reaction conditions [6,7]. MBH adducts such as acetates, bromides and carbonates are important synthons in the synthesis of biologically-active compounds and natural products [8]. Despite all these advantages, there are still limitations of MBH reactions of some substrates such as α -keto esters. Basavaiah et. al. reported the MBH reaction of α -keto esters with alkyl vinyl ketones and cyclic enones mediated by titanium tetrachloride [9, 10]. An organocatalytic version of the MBH reaction between cyclopent-2-enone and α -keto esters was reported by Shi and Zhang catalyzed by diphenylmethylphosphine and phenol additives [11]. However, the organocatalytic MBH reaction of α -keto esters remains unexplored.

In recent years, computational studies have also been performed to understand the MBH reaction mechanism and probable transition states to design more efficient organocatalysts [12-15]. We concentrated on the alcohol-mediated MBH mechanism proposed by Aggarwal as outlined in Scheme 2 [16]. According to the experimental and computational studies, the rate-limiting step of the reaction is the methanol-aided hydrogen transfer between intermediate 2 (int2) and 3 through the corresponding transition state (TS).



Scheme 2. Mechanism of the alcohol-mediated MBH reaction [16]

Amino acid/NR₃ co-catalyst systems are very common in asymmetric MBH reactions and promising results have been obtained with L-proline [17-19]. L-Pipecolinic acid has been used as a co-catalyst with *N*-methylimidazole in the asymmetric intramolecular MBH reaction by Aroyan et al. [20].

We evaluated *N*-Boc-L-amino acids as a mediator to transfer hydrogen between intermediate 2 and 3 (Scheme 2) in the tertiary amine catalyzed MBH reaction of methylphenyl glyoxylate with methyl vinyl ketone in *N,N*-dimethylformamide (DMF).

2. Materials and Methods

All chemicals and solvents were used as received from commercial suppliers and used without further purification. Silica gel F254 (Merck 5554) precoated plates were used for thin-layer chromatography (TLC). Infrared spectra were recorded on a Thermo-Nicolet 6700 FTIR. NMR experiments were performed on a Varian Mercury Plus 400 MHz spectrometer at ambient temperature using TMS as an internal standard. The elemental analyses were performed on a Costech ECS 4010 analyzer. Melting points were determined in open glass capillary tubes with an Electrothermal digital melting point apparatus.

2.1. General Procedure for the MBH Reaction

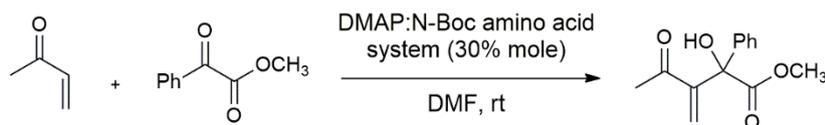
A mixture of methyl vinyl ketone (0.6 mmol, 0.05 ml), (0.3 mmol, 0.04 ml), 4-dimethylaminopyridine (DMAP) (0.1 mmol, 12 mg), *N*-Boc protected amino acid catalyst (0.1 mmol) in DMF (0.5 ml) was stirred at room temperature. The reaction progress was monitored with TLC until methylphenyl glyoxylate was consumed. DMF was evaporated under vacuum and the crude product was purified with column chromatography.

2.1.1. Methyl 3-hydroxy-4-methylene-2,5-dioxo-3-phenylhexanoate

Colorless crystal, 66% yield, m.p. 76-77 °C. FT-IR, ν , cm⁻¹: 3242 (O-H stretching), 2879 (C-H stretching), 1766 (C=O stretching), 1638 (C=C stretching). ¹H-NMR (400 MHz, CDCl₃) δ ppm: 2.41 (s, 3H), 3.77 (s, 3H), 4.84 (bs, 1H), 5.54 (s, 1H), 6.22 (s, 1H), 7.33-7.39 (m, 3H), 7.56-7.58 (d, *J* = 8.0 Hz, 2H). ¹³C-NMR (101 MHz, CDCl₃) δ ppm: 26.40, 53.18, 77.24, 78.73, 126.63, 128.26, 128.33, 129.68, 137.61, 150.86, 174.01, 200.45. Anal. calcd. for C₁₃H₁₄O₄: C 66.66; H 6.02. Found: C 66.61; H 6.00.

3. Results and Discussion

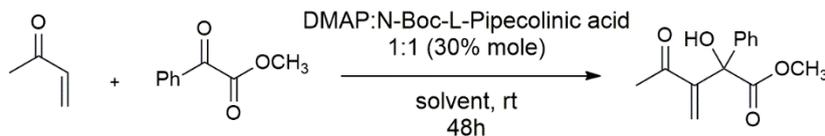
We used a standard MBH reaction of methylphenyl glyoxylate with methyl vinyl ketone in DMF to determine the activities of *N*-Boc-protected amino acid catalysts (Table 1). Both amino acids and DMAP were inactive when they were used alone.

Table 1. Standard MBH reaction and structures of the catalysts

Catalyst	Time (h)	Yield (%) ^a
 Boc-L-proline (1)	48	33
 N-Boc- <i>trans</i> -4-hydroxypyrrolidine-2-carboxylic acid (2)	96	16
 N-Boc-L-pipecolinic acid (3)	48	66
 (1R,3S,4S)-N-Boc-2-azabicyclo[2.2.1]heptane-3-carboxylic Acid (4)	72	23

^a Isolated yields

The catalytic activities of four different cyclic and N-Boc-protected L-amino acids were investigated. The best results were obtained with N-Boc-L-pipecolinic acid (**3**) with 66% yield in 48 hours at room temperature. We performed chiral HPLC analysis to check the enantioselectivity of the reactions, as the catalysts were chiral. However, all of the products were obtained in a racemic form. Then, we performed the reaction in different solvents to optimize the conditions (Table 2).

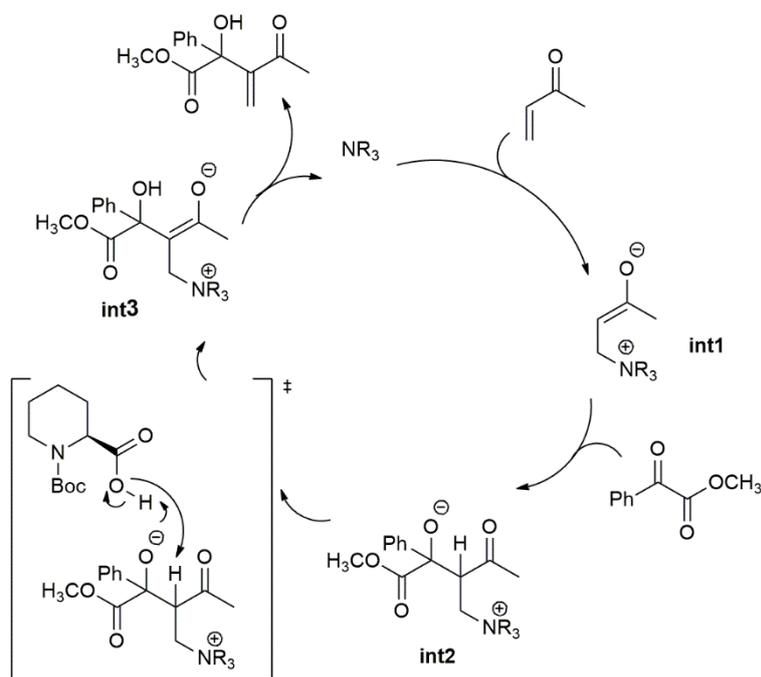
Table 2. Solvent optimization

No	Solvent	Yield ^a
1	DMF	66
2	MeOH	trace
3	EtOH	trace
4	THF	-

^a Isolated yields

The best results were obtained in DMF and only a trace amount of product formation was observed in alcohols such as methanol and ethanol while no product was obtained in tetrahydrofuran (THF).

According to these results, we also proposed a reaction mechanism mediated by **3** (Scheme 3).

**Scheme 3.** Proposed reaction mechanism for N-Boc-L-pipecolinic acid (**3**) mediated MBH reaction of methylphenyl glyoxylate

The failure of the catalytic system in terms of enantioselectivity was also explained by the role of **3** in the proposed reaction mechanism.

4. Conclusions

We developed a catalytic system for the MBH reaction of methylphenyl glyoxylate with methyl vinyl ketone. The recently-proposed reaction mechanisms revealed the important role of a proton transfer mediator in this reaction. We

designed our catalytic system according to the results of these mechanistic studies. We successfully performed the reaction with DMAP and N-Boc-L-pipecolinic acid. We also proposed a proper reaction mechanism. Our ongoing studies are related to the expansion of the substrate scope and the asymmetric version of this reaction.

5. Acknowledgments

G. K. thanks TUBITAK for the fellowship (TUBITAK-2218). Bursa Technical University, Scientific Research Fund is gratefully acknowledged for financial support (Project No: 182N02).

References

- [1] Shi, M. and Liu, Y.-H. (2006). Traditional Morita–Baylis–Hillman reaction of aldehydes with methyl vinyl ketone co-catalyzed by triphenylphosphine and nitrophenol. *Org. Biomol. Chem.*, 4; 1468-1470.
- [2] Basavaiah, D., Rao, A. J., Satyanarayana, T. (2003). Recent Advances in the Baylis–Hillman Reaction and Applications. *Chem. Rev.*, 103(3); 811- 892.
- [3] Basavaiah, D. and Naganaboina, R. T. (2018). The Baylis–Hillman reaction: a new continent in organic chemistry-our philosophy, vision and over three decades of research. *New J. Chem.*, 42; 14036-1406.
- [4] Shukla, P., Asati, A., Patel, D., Singh, M., Rai, V. K., Rai, A. (2023). Novel Synergistic Catalysis by Ethylcarbodiimide Hydrochloride Salt and CuI Towards Morita-Baylis-Hillman Reaction. *ChemistrySelect*, 8; 571-574.
- [5] Wang, C.-C. and Wu, X.-Y. (2011). Catalytic asymmetric synthesis of 3-hydroxyl-2-oxindoles via enantioselective Morita–Baylis–Hillman reaction of isatins. *Tetrahedron*, 67; 2974-2978.
- [6] Crawshaw, R., Crossley, A. E., Johannissen, L., Burke, A. J., Hay, S., Levy, C., Baker, D., Lovelock, S. L., Green, A. P. (2022). Engineering an efficient and enantioselective enzyme for the Morita-Baylis-Hillman reaction. *Nature Chemistry*, 14; 313-320.
- [7] Maity, S. and Szpilman, A. M. (2023). 2-Fluoroenones via an Umpolung Morita–Baylis–Hillman Reaction of Enones. *Org. Lett.*, 25(7); 1218-1222.
- [8] Reddy, T. N., and Rao, V. J. (2018). Importance of Baylis-Hillman adducts in modern drug discovery. *Tetrahedron Letters*, 59; 2859-2875.
- [9] Basavaiah, D., Sreenivasulu, B., Reddy, R. M., Muthukumaran, K. (2001). The Baylis-Hillman Reaction: TiCl₄ Mediated Coupling of Alkyl Vinyl Ketones with α -Keto Esters and Aldehydes. *Synthetic Communications*, 31(19); 2987-2995.
- [10] Basavaiah, D., Sreenivasulu, B., Rao, A. J. (2003). Steric Factors Direct Baylis-Hillman and Aldol Reactions in Titanium Tetrachloride Mediated Coupling between α -Keto Esters and Cyclohex-2-enone Derivatives. *J. Org. Chem.* 68; 5983-5991.
- [11] Shi, M. and Zhang, W. (2005). Organocatalysts of tertiary-phosphines and amines catalyzed reactions of α -keto esters with cyclopent-2-enone. *Tetrahedron*, 61; 11887-11894.
- [12] Wei, Y. and Shi, M. (2013). Recent Advances in Organocatalytic Asymmetric Morita–Baylis– Hillman/aza-Morita–Baylis–Hillman Reactions. *Chem. Rev.*, 113(8); 6659-6690.

- [13] Price, K. E., Broadwater, S. J., Jung, H. M., McQuade, D. T. (2005). Baylis-Hillman mechanism: a new interpretation in aprotic solvents. *Org. Lett.*, 7; 147-150.
- [14] Liu, Z, Patel, C., Harvey, J. N., Sunoj, R. B. (2017). Mechanism and reactivity in the Morita-Baylis-Hillman reaction: the challenge of accurate computations. *Phys. Chem. Chem. Phys.*, 19; 30647-30657.
- [15] Seumer, J., Hansen, J. K. S., Nielsen, M. B., Jensen, J. H. (2023). Computational Evolution of New Catalysts for the Morita–Baylis– Hillman Reaction. *Angew. Chem. Int. Ed.*, 62; e202218565.
- [16] Robiette, R., Aggarwal, V. K., Harvey, J. N. (2007). Mechanism of the Morita-Baylis-Hillman Reaction: A Computational Investigation. *J. Am. Chem. Soc.*, 129; 15513-15525.
- [17] Tang, H., Zhao, G., Zhou, Z., Tang, Q. Z. C. (2006). Synthesis of some new tertiary amines and their application as co-catalysts in combination with L-proline in enantioselective Baylis-Hillman reaction between o-Nitrobenzaldehyde and methylvinylketone. *Tetrahedron Lett.*, 47; 5717-5721.
- [18] Chen, S.-H., Hong, B.-C., Su, C.-F., Sarshar, S. (2005). An unexpected inversion of enantioselectivity in the proline catalyzed intramolecular Baylis-Hillman reaction. *Tetrahedron Lett.*, 46; 8899-8903.
- [19] Shi, M., Jiang, J.-K., Li, C.-Q. (2002). Lewis base and L-proline co-catalyzed Baylis-Hillman reaction of arylaldehydes with methyl vinyl ketone. *Tetrahedron Lett.*, 43(1); 127-130.
- [20] Aroyan, C. E., Vasbinder, M. M., Miller, S. J. (2005). Dual Catalyst Control in the Enantioselective Intramolecular Morita-Baylis-Hillman Reaction. *Org. Lett.* 7(18); 3849-3851.

Preventing Crime and Terrorist Activities with a New Anomaly Detection Approach Based on Outfit

Gizem ORTAC KOSUN^{1*} , Seckin YILMAZ² , Yusuf KAYIPMAZ² , Rüya SAMLI¹ 

^{1*} Computer Engineering Department, Istanbul University - Cerrahpasa, 34320, Bursa, TURKEY

² Computer Engineering Department, Bursa Technical University, 16310, Bursa, TURKEY

Abstract

Video surveillance systems play an important role in ensuring security indoors and outdoors and detecting suspicious persons due to the increasing violence and terrorist acts every year. In the proposed study, an artificial intelligence-based warning system has been developed, which enables the detection of potential suspects who may carry out criminal or terrorist activities by detecting anomalies in surveillance videos. In this developed system, an abnormality is detected by using the outfits of the people. The YOLOv7 object detection model is trained on our customized data sets, and suspicious person detection is made through outfit information. Especially in cases where biometric data is hidden, dress information makes it easier to obtain information about people. For this reason, the knowledge of outfits is the main point of this study in the detection of suspicious persons. Thanks to this study, security guards will be able to focus on this suspicious person before they pre-empt any crime or terrorist activity. If there are other data confirming the suspicious situation as a result of this follow-up; security personnel will have time to eliminate the crime or attack. The experimental results obtained have been promising in terms of the usability of a person's outfit anomalies to ensure public confidence or avoid risk to human life. Although there are various studies in the literature for the prevention of terrorist or criminal activities; there is no study in which people's outfit is used to identify suspects.

Keywords: Forensic science, Anomaly detection, Soft biometrics, Surveillance video.

Cite this paper as:

Ortac Kosun, G., Yilmaz S., Kayıpmaz Y. and Samli, R. (2023). *Preventing Crime and Terrorist Activities with a New Anomaly Detection Approach Based on Outfit*. Journal of Innovative Science and Engineering, 7(2):167-182

*Corresponding author: Gizem Ortac Kosun

E-mail: gizemortac1@hotmail.com

Received Date:22/08/2023

Accepted Date:08/10/2023

© Copyright 2023 by

Bursa Technical University. Available online at <http://jise.btu.edu.tr/>



The works published in Journal of Innovative Science and Engineering (JISE) are licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

1. Introduction

With the increase in terrorism, violence, and crime rates in recent years, the importance of security and safety issues is increasing day by day. Government and private organizations are concerned about security in public and crowded areas such as airports and shopping malls. For this reason, video surveillance systems have been used in both private and public places. In forensic investigations, video and images from these systems are widely used in crime evidence investigations, which can provide important elements of forensic evidence, bring together existing elements of evidence, or make connections between evidence in a particular case [1]. However, most existing surveillance systems rely on the human factor. The efficiency of these systems, which rely on human control to detect any abnormality, decreases over time. This problem can be solved by the automation of video surveillance. These artificial intelligence-based studies have gained great importance in facilitating control operations with video surveillance systems and reducing the human factor error rate. The function of the automated system is to give an indication, in the form of an alarm or in any other form, when pre-defined abnormal activity occurs [2]. The intended use of the recorded images is to identify potential suspects or take appropriate action if they have no knowledge of where and when the incident occurred or even whether it occurred [3]. However, many of the systems, especially those originating from crime or terrorism, are designed to facilitate the investigation process after the event has occurred. Although the number of studies in this area is limited; focused on detecting offensive tools such as guns and knives, and developing preventive and warning systems. However, these tools, which constitute an element of attack, are usually hidden by people in outfits such as coats and revealed when the person approaches the crime environment. Studies show that the use of dress codes is very effective for detecting suspects, criminals, or missing persons [4,5]. Therefore, different approaches are needed to identify potential suspects.

In this study, an artificial intelligence-based warning system is proposed to help security guards identify potential suspects. It is focused on outfits to prevent crime or terrorist activities in video surveillance systems. In this developed system, if there is an anomaly in outfits worn by the people, early warning is given about the suspicious situation. Thus, security personnel will focus on the relevant person and follow the person closely with advance warning. As a result of this close follow-up monitoring, if there is any other data confirming the suspicious individual situation, the security personnel will have time to eliminate the crime or terrorist attack. Before the attack on the United States Capitol Hill building on January 6, 2021, video footage was obtained showing the route the bomber walked on January 5, 2021, wearing the bomber's face mask, glasses, hoodie, and gloves. Owing to these images, it was determined that the bomber placed the devices in the alley behind the Republican National Committee Headquarters between 19.30 - 20.30 in the evening [6]. This study is the first proposed study in this direction in the literature.

Based on the real events experienced, the study was advanced through various scenarios. For example, it is abnormal for a person to wear a coat on a summer day. If this person approaches the entrance of a public or private establishment, there may be a potential risk of attack. In such a case, this proposed system will provide a warning via the security camera while these subject is moving toward the entrance. Thus, alerting the security guard and being able to take precautions before the person approaches the building. To give another example, if a person's face is hidden among the crowd in public places such as shopping malls, a warning will be given that there is a potential risk of posing a threat to that person. In this way, the security guard will be able to follow this person closely. As a result of the close monitoring

of the security personnel, if there is a possibility to prevent a crime before it is committed, it can be prevented. If a crime has taken place (such as theft), it will be possible to take action much more quickly to catch this person. Otherwise, there will be situations where security personnel cannot detect people who may raise such suspicion during long-term monitoring of the security camera due to eye strain.

This proposed system, makes significant contributions to forensic prevention, catching the criminal, or clarifying the cases. The rest of the work is organized as follows: Part 2 is related studies, Part 3 is the methodology, Part 4 is experimental results, and Part 5 is the conclusion and acknowledgment.

2. Related Works

Researchers have shown great interest in studies involving the detection of various anomalous objects and situations, such as people, anomalies, and masks, within the framework of forensic events in video systems. With these studies, it has become much easier to collect and analyze important evidence about cases in video systems. Detecting suspicious situations, people, or objects and developing warning systems has been an area of great interest in the past years. In the study of Narejo et al.[7], a computer-based fully automatic system was developed to detect various weapons, especially pistols, and rifles. Using the YOLOv3 algorithm on customized datasets, a model is proposed that provides a predictive machine or robot to identify weapons and can also alert the human manager when a gun or firearm is seen on the sidelines. The system has an accuracy value of 98.89%. By Grega et al. [8], a system was developed that detects knives and firearms in CCTV images and warns the security guard or operator. The specificity and sensitivity of the blade detection algorithm are 94.93% and 81.18%, respectively, when the edge histogram descriptor and the Decision Support Vector classifier are used. These results are significantly better than others published recently. A specificity of 96.69% and a sensitivity of 35.98% were obtained by using a three-layer neural network for the firearm detection algorithm. Mehta et al. [9] developed a real-time warning system that includes weapon and fire detection anomalies in areas monitored by cameras. Experimental results have shown that the proposed model is suitable for real-time monitoring and can be deployed in any GPU-based system. Marbach et al. [10] proposed a system for automatic fire detection based on the temporal variation of fire density. Dever et al. [11] designed an armed robbery detection algorithm according to the silhouette of the person and the position of the arms. The silhouette image was divided into separate parts and the position of the arms was determined and the determination was made. Anomaly detection in crowded areas is also one of the areas of increasing interest for public safety in video surveillance systems. However, although many studies have been carried out in this area in recent years such as Yin et al. [12], Ravanbakhsh et al. [13] and Mehran et al. [14] the subject is still an open field of study.

3. Material and Method

The flow chart of the study is presented in Figure 1. After the collected training images went through the preprocessing stage, the regions of the objects to be detected in the image were tagged and trained with the YOLOv7 network. In the test phase, images containing suspicious situations were applied to the trained YOLOv7 model, results were obtained and a warning was given in case of suspicious situation detection.

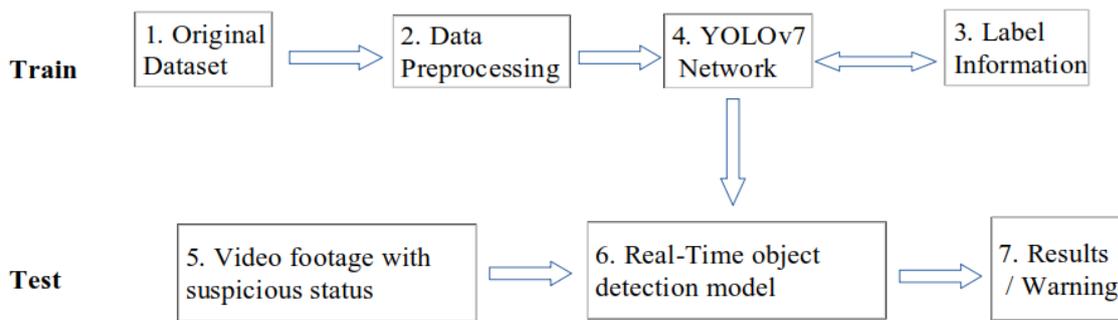


Figure 1. Flow chart of the proposed algorithm.

3.1. Datasets

In the proposed study, an algorithm is proposed to detect situations where people camouflage the face and trunk of their bodies in a way that prevents recognition and tracking. The human body is divided into two parts, the head and the trunk. It is aimed to identify people who hide some or all of their biometric data by wearing a surgical mask, prescription glasses, and sunglasses on the head area, and a coat, overcoat, or coat in a size that will cover their body regardless of the weather conditions, and to detect possible dangers in advance.

3.2. Data Collection

In the proposed study, the surgical mask, transparent glasses, and sunglasses are required for fixation in the head and trunk region; The images required for coats, coats, overcoats, and clothes other than those outfits were obtained from Google Images. There are 220 images in the first dataset to be used for coat detection, and 800 images in the second dataset that will be used in the detection of surgical masks, clear glasses, and sunglasses.

3.3. Data Preprocessing

After the necessary data has been obtained and collected, the next step is to annotate the data. This step has a direct impact on the efficiency and performance of the model. LabelImg \cite{heartexlabs}, a graphic image annotation tool, was used to label the ground truth box of images. Label categories are divided into “mask”, “no mask”, “glass”, and “sunglass” for the head area, and “coat” and “other” for the body area, respectively. The dataset is annotated in the format $(\langle x_{min} \rangle \langle y_{min} \rangle \langle x_{max} \rangle \langle y_{max} \rangle)$ and converted to YOLO format $(\langle object-class \rangle \langle x_{center} \rangle \langle y_{center} \rangle \langle width \rangle \langle height \rangle)$.

For the head area of the people, 800 images are selected in categories: "mask", "no-mask", "glass" and "sunglass", considering whether people are masked at different angles, positions, and directions, whether they wear clear glasses and sunglasses. manually labeled. of the tagged photos are reserved for the training set and the validation set. During the labeling phase, the people in the images were labeled as wearing or not wearing a mask to cover their entire faces. If the person wears clear glasses, in addition to the mask information label, the glasses are labeled to cover the glasses. If the person wears sunglasses, a sunglasses label is given to cover the sunglasses in addition to the mask data. In the part of the study that included coats, and topcoats (t-shirts, sweaters, athletes, etc.), 220 images were labeled as "coat" and "other" according to whether the person wore them or not. 190 of these photographs were used as a training

set and as a validation set.

3.4. Method

In this proposed system, when such a situation occurs, a warning will be given over the system while the person is moving toward the entrance. In this way, the security guard will be able to take precautions without getting too close to the building. The person's face recognition process is also very important for detecting the potential criminal. In this subject, the human body was divided into two parts the head and the torso, and experiments were carried out on the detection of abnormal situations. Masks, glasses, and sunglasses that prevent face recognition in the head area were detected. In the body part, the outfit will prevent recognition and explosives, weapons, knives, etc. It is aimed to determine whether a coat, overcoat, or overcoat is worn to hide objects. This study makes significant contributions to the prevention of forensic incidents, catching criminals, and illuminating cases. For this system, where real-time detection and speed are important, the most up-to-date version of You Only Look Once (YOLO) [16] algorithm, which is the fastest and most accurate among object recognition algorithms, is used.

YOLO stands for "You Only Look Once," an open-source object detection algorithm that relies on convolutional neural networks.

It's renowned as one of the most well-known deep learning algorithms, primarily due to its remarkable speed, attributed to its single-stage detection design. Traditional detection systems utilized classifiers or localizers for identifying objects, unlike YOLO, which takes a distinct approach. YOLO treats object detection as a regression task, employing a sole neural network for the entire image. In the YOLO framework, the input image is initially divided into a grid of size $S \times S$. Each grid cell is responsible for determining the presence of an object within its region, pinpointing its center, checking if it spans its center, measuring its dimensions (length and height), and classifying it. These processes culminate in the creation of bounding boxes [17].

Over the years, various iterations of the YOLO algorithm have been introduced by researchers. The lineage extends from YOLOv1 to the widely recognized YOLOv3, originating from the efforts of Joseph Redmon, a graduate student, and Ali Farhadi, a consultant. Following Redmon's withdrawal from computer vision research due to ethical concerns, Alexey Bochkovskiy introduced YOLOv4, carrying the torch forward. YOLOv7, the latest official version of the YOLO architecture, has been developed by its original creators.

The YOLOv7 algorithm has garnered significant popularity within the computer vision and machine learning communities as a potent object detection method. It eclipses all preceding object detection models and YOLO versions in both swiftness and precision. Notably, it operates efficiently on more budget-friendly hardware than many other neural networks and can be swiftly trained on modest datasets without relying on pre-trained weights. Taking into account the imperative of real-time detection, the YOLOv7 model, the most recent advancement in the YOLO series, has been selected, as illustrated in Figure 2.

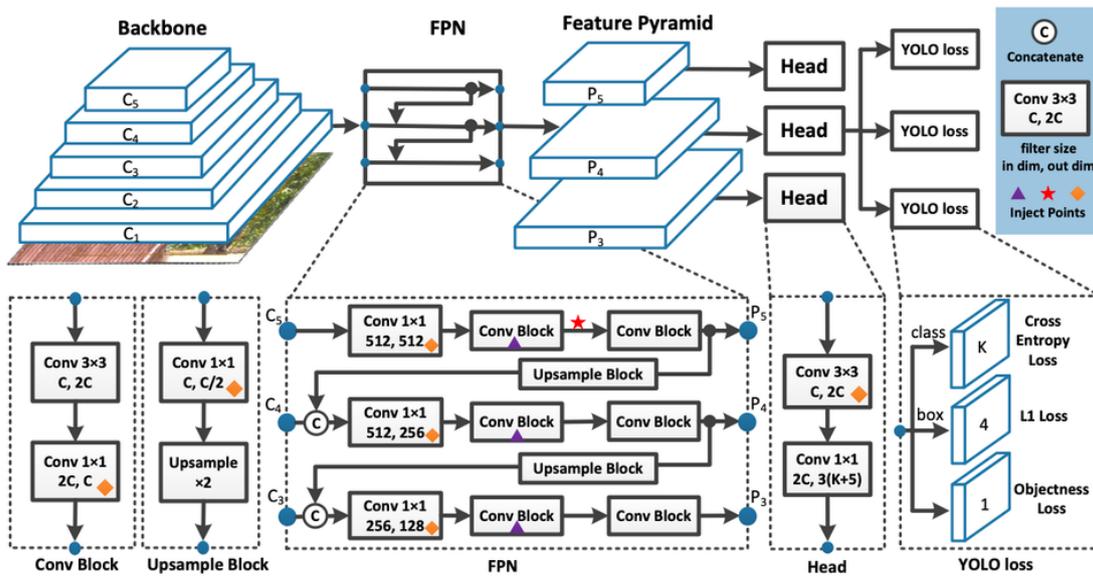


Figure 2. YOLOv7 architecture [18].

The YOLOv7 model has elevated both the pace and precision of object detection compared to its previous iterations. It introduces an enhanced integration approach, delivers more exact object detection performance, incorporates a more resilient loss function, and boasts an improved network architecture that features an upgraded tag assignment function. In a general sense, all YOLO architectures comprise three main components: the spine, head, and neck. The spine takes on responsibilities such as foundational work and essential feature extraction, channeling this information to the head through the intermediary neck component.

Unlike its predecessors, YOLOv7 adopts an extended and efficient layer aggregation network (E-ELAN) as the computational block for its backbone, refraining from deviating and continuing to employ DarkNet for the backbone [19]. To augment the network's capacity for continual learning enhancement without disrupting the original gradient pathway, YOLOv7 integrates a long-range attention network termed Extended-ELAN (abbreviated as E-ELAN). This innovation employs principles of expansion, mixture, and unification to bolster learning capabilities [18].

In terms of architecture, E-ELAN introduces alterations exclusively within the computational block, while the architecture of the transition layer remains largely unchanged. The strategic approach involves leveraging group convolution to broaden the channel and materiality of the computational blocks, as depicted in Figure 3 [20].

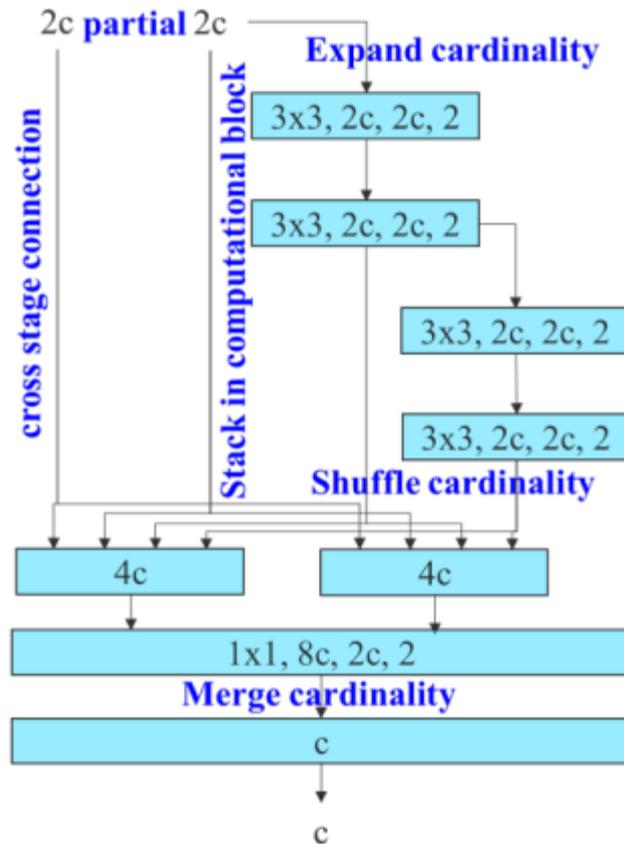


Figure 3. Extended ELAN (E-ELAN)

The YOLOv7 algorithm provides reparameterization planning (RP). RP is based on averaging several models to produce a performance-solid final model. Module-level reparameterization has been an active area of research where certain parts of the model have specific reparameterization strategies [20]. After training several models with the same parameters and different training sets, the weights are averaged to obtain the final model. The final model is formed by combining the outputs. The reparameterized convolution architecture in YOLOv7 uses RepConv [21] without the identity link (RepConvN). The goal is to prevent identity bindings when reparameterized convolution is used to replace a convolution layer with residue or concatenation.

Training Environment and Training Parameters

The experiments described in this paper were carried out using a 64-bit CPU operating at 2.20 GHz with twelve cores, along with 8 GB of memory. The computing setup also included an NVIDIA GeForce RTX 3050 Laptop GPU with 4 GB of video memory. The utilized version of the Compute Unified Device Architecture (CUDA) was 11.03. The deep learning framework of choice was PyTorch 1.11.0, and the code was compiled using Python 3.8.

Within the experimental model outlined in this article, the Adam optimization algorithm was deliberately employed to enhance the pace of training. The model processes images with a resolution of 640×640 pixels as its input. The initial learning rate assigned to the model is 0.01, coupled with a learning rate momentum of 0.937, and a weight decay value of 0.0005. These parameter choices consider both training speed and video memory capacity.

In terms of the training process, the batch size for each training set within the study was fixed at 4 samples. The model underwent 100 rounds of training iterations, each consisting of an equivalent number of iterations.

Evaluation Metrics

The evaluation of our model's performance in this article is carried out comprehensively and objectively using metrics such as the confusion matrix, precision, and sensitivity values.

TP represents the correct perception that the model predicts positively and the true value is also positive. FN detects detection errors that the model predicts negatively but have a positive true value. FP represents the detection errors that the model predicts positively but has a negative true value. TN represents accurate detection; the model estimate is negative and the true value is also negative.

Expressions of precision and sensitivity are calculated as follows:

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

As precision and recall alone may not suffice as exclusive performance indicators for the model, the F1 score is introduced as a compromise metric that combines both aspects, as defined in the following formula [22]:

$$F1 = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (3)$$

The model's performance was assessed using a PR curve, which takes into account the precision and sensitivity ratio for each detected category. Additionally, the mean precision (mAP) is employed as a measure of the model's accuracy, depending on precision and recall. In this context, (AP) refers to the area under the precision-recall curve (PR curve), and (mAP) represents the average of (AP) values across various classes. Here, (N) represents the number of classes within the test sample [23].

$$AP = \int_0^1 P(R) dR \quad (4)$$

$$mAP = \frac{\sum_1^N \int_0^1 P(R) dR}{N} \quad (5)$$

Loss Function

The loss function of the YOLOv7 model is calculated by summing up three loss values: loss of localization (L_{box}), loss of confidence (L_{obj}), and loss of classification (L_{cls}). Among them, loss of confidence and loss of classification functions, binary cross-entropy loss, and localization loss use the CIoU loss function.

$$\text{Loss} = W1 \times L_{\text{box}} + W2 \times L_{\text{cls}} + W3 \times L_{\text{obj}} \quad (6)$$

In this context, the weight values for the three loss functions are denoted as W1, W2, and W3, respectively [22].

For the coordinate loss, the established CIoU loss [24] is utilized, which factors in considerations like overlapping area, center distance, and aspect ratio. This loss contributes to enhanced detection accuracy, particularly addressing the issue of non-overlapping detection boxes.

The computation formula for the BCE cross-entropy loss is defined as follows, where w_n denotes the averaging of outcomes, and y_n signifies the actual sample label:

$$L_n = -w_n \cdot \left[y_n \cdot \log(S(x_n)) + (1 - y_n) \cdot (1 - \log(S(x_n))) \right] \quad (7)$$

The calculation formula for the CIoU loss is specified as follows, where IoU denotes the intersecting area between the prediction box and the actual box:

$$\text{CIoU} = \text{IoU} - \left(\frac{\rho^2(b, b^{\square})}{b^2} + \alpha v \right) \quad (8)$$

The parameter v quantifies the alignment of the detection frame's aspect ratio, while the parameter α serves as a trade-off parameter, allowing for a greater emphasis on regressing the overlapping area factor [23].

4. Results and Discussion

The YOLOv7 model is trained on two datasets. Loss of localization, loss of confidence, loss of classification, precision, recall, and graphs are presented for both datasets during the training period of the training and validation sets. The loss types included in these graphs are object loss, classification loss, precision, sensitivity, and values. The localization loss indicates how well the algorithm can detect the center of an object and how well the estimated bounding box covers an object. Confidence is a measure of the probability that an object will exist in a suggested area of interest, and high objectivity means that the viewport likely contains an object. Classification loss gives information about how well the algorithm can predict the correct class of a given object.

Different performance measures for both the training and validation sets of the first data set consisting of the “coat” and “other” classes are presented in Figure 4. As can be seen from the graphs, the precision, sensitivity, and average precision values developed rapidly throughout the training. On the other hand, it is seen that localization loss, confidence loss, and classification loss decrease rapidly throughout the training.

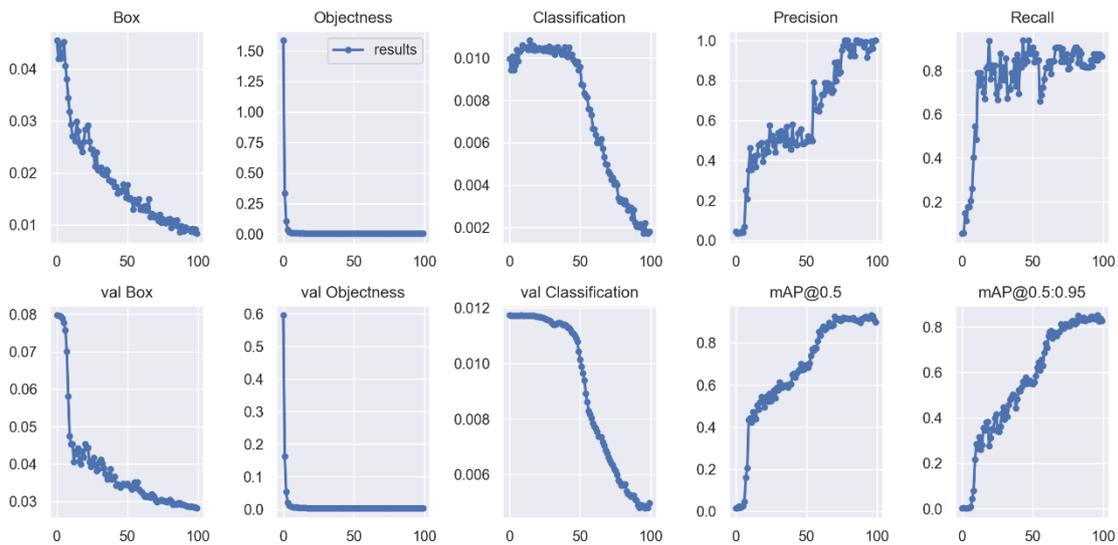


Figure 4. Loss of localization, loss of confidence, loss of classification, precision, sensitivity and mAP plots during training for the first dataset training and validation set.

For the first dataset, 100 rounds of training were completed in approximately 0.73 hours. As can be seen in Table 1, the precision of the dataset trained on YOLOv7 is 0.865, mAP@0.5 is 0.898, and mAP@0.5:0.95 is 0.827, indicating that the performance in the validation set is quite high.

Table 1. First data set validation performance values.

Class	Precision	Recall	mAP@.5	mAP@.5:.95
Coat	1	0.865	0.898	0.827
Other	1	1	0.996	0.938
All	1	0.729	0.779	0.717

As can be seen from the F1 curve presented in Figure 5, the confidence value optimizing precision and sensitivity is 0.702. A high confidence value indicates that a suitable design has been obtained.

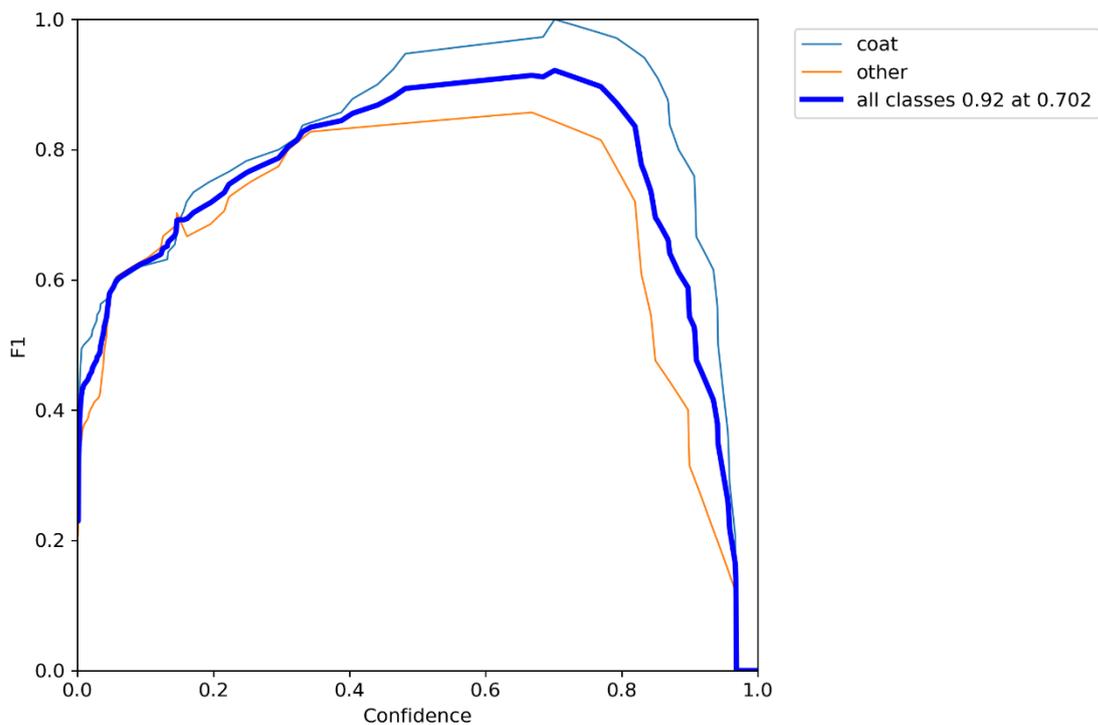


Figure 5. F1 curve for the first data set.

After the model was trained, predictions were made for new and invisible images in the test set. The examples in Figure 6 show that the algorithm can detect the person wearing the coat with greater precision.

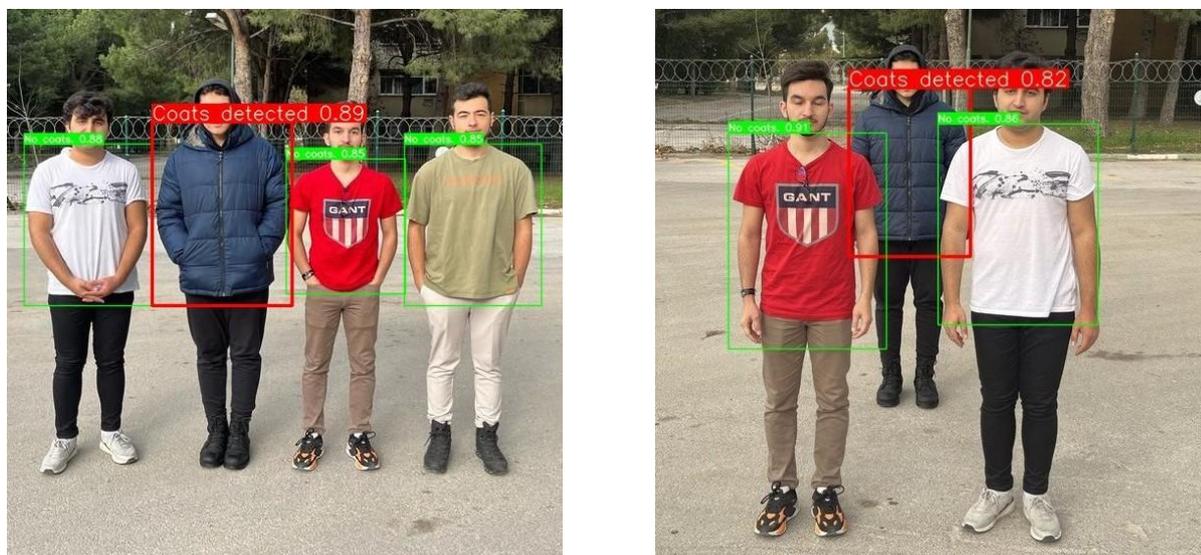


Figure 6. An example test set result for the first data set.

Different performance measurements for both training and validation sets of the second data set, consisting of “no-mask”, “mask”, “glass” and “sunglass” classes, are presented in Figure 7. As can be seen from the graphs, the precision, sensitivity, and average precision values developed rapidly throughout the training. On the other hand, localization loss, confidence loss, and classification loss showed a rapid decrease for the training set throughout the training. However, it is seen that the classification and confidence loss increase for the validation set.

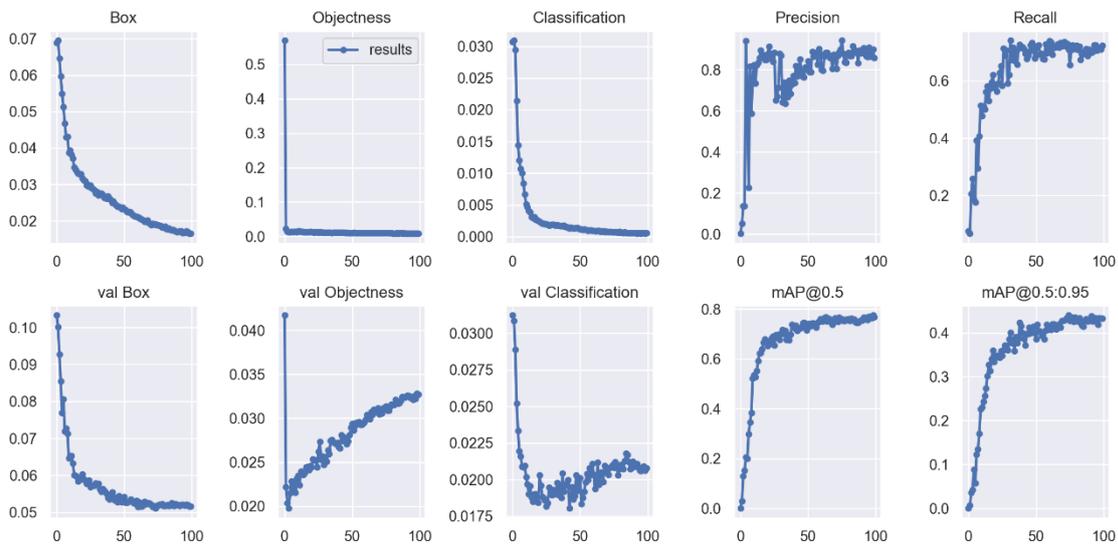


Figure 7. Graphs of loss of localization, loss of confidence, loss of classification, precision, sensitivity and mAP during training for the second dataset training and validation set.

For the second dataset, 100 rounds of training were completed in approximately 2.6 hours. As can be seen in Table 2, the precision of the dataset trained on YOLOv7 is 0.858, the sensitivity ratio is 0.724, mAP@0.5 is 0.768, mAP@0.5:0.95 is 0.433, and the performance in the validation set is quite high.

Table 2. Second data set validation performance values.

Class	Precision	Recall	mAP@.5	mAP@.5:.95
no-mask	0,764	0,824	0,846	0,451
mask	0,891	0,925	0,946	0,593
glasses	0,862	0,814	0,833	0,419

As can be seen from the F1 curve presented in Figure 8, the confidence value optimizing precision and sensitivity is 0.539. Confidence value shows that a suitable design is obtained.

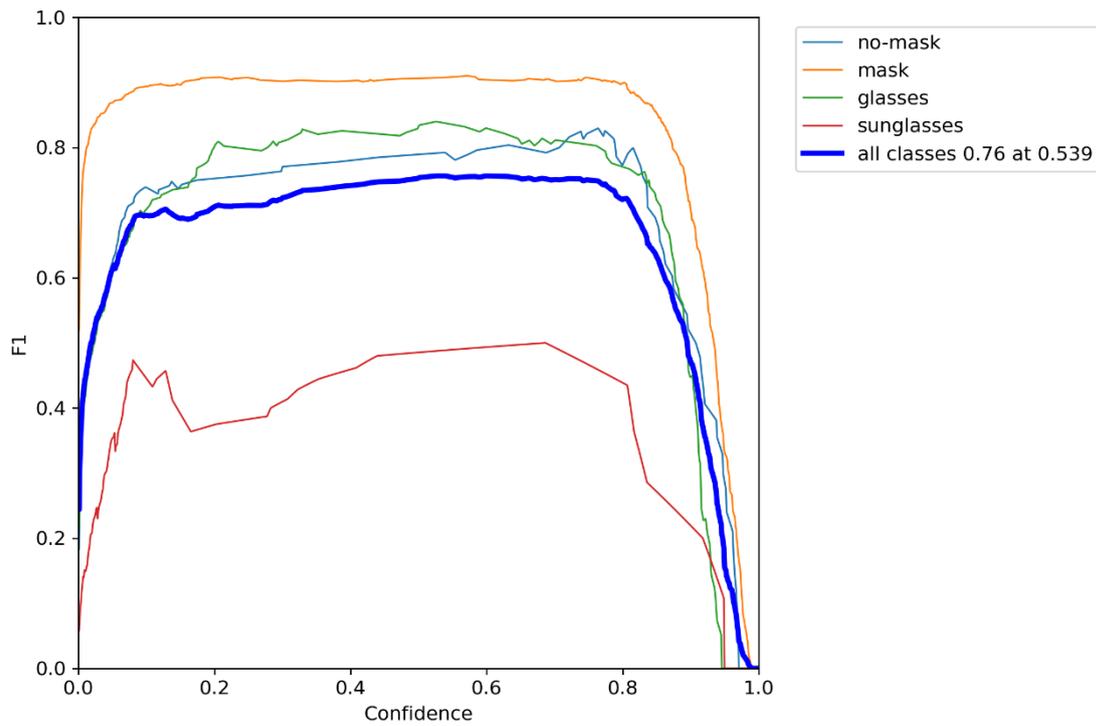


Figure 8. F1 curve for the second data set.

After the model was trained, predictions were made for new and unseen images in the test data set. The use of facial recognition and biometric data is an approach that increases performance in various studies such as person recognition and tracking, emotion and expression recognition, and social behavior analysis. Facial features such as eyes, nose, and mouth carry important information for face recognition. When the mask is worn, the nose and mouth are covered and face recognition cannot be performed. Wearing a mask disrupts the holistic face processing that supports face detection and recognition. As a result, face-matching performance deteriorates, making it difficult to track people. Besides this, people wearing sunglasses prevent the identification process if the eyes are closed. As can be seen from the results in Figure 9 and Figure 10, people who wear masks, sunglasses, or glasses to hide their biometric data can be detected at a high rate. Information alerts are created for cases where the mouth of people wearing masks and the eye area of people wearing sunglasses cannot be detected.



Figure 9. An example test set result for the second data set.



Figure 10. An example test set result for the second data set.

developed a catalytic system for the MBH reaction of methylphenyl glyoxylate with methyl vinyl ketone. The recently-proposed reaction mechanisms revealed the important role of a proton transfer mediator in this reaction. We designed our catalytic system according to the results of these mechanistic studies. We successfully performed the reaction with

DMAP and N-Boc-L-pipecolinic acid. We also proposed a proper reaction mechanism. Our ongoing studies are related to the expansion of the substrate scope and the asymmetric version of this reaction.

5. Conclusion

In the current study, an algorithm that detects outfit anomalies is proposed. The algorithm was developed with the YOLOv7 method. With this method, which provides high performance in real-time object detection, abnormal situations have been detected quite successfully by using some accessories and outfits. The results obtained from the experimental studies have been promising in establishing an early warning system for the detection of suspicious persons by testing them on scenarios applicable to daily life.

In future studies, the scope of outfits and accessories that people will use to hide their identities will be expanded. In addition, a more robust model will be developed for different camera positions, long distances, and more crowded environments.

References

- [1] Xiao, J., Li, S., & Xu, Q. (2019). Video-based evidence analysis and extraction in digital forensic investigation. *IEEE Access*, 7, 55432-55442.
- [2] Kamthe, U. M., & Patil, C. G. (2018, August). Suspicious activity recognition in video surveillance system. In 2018 Fourth international conference on computing communication control and automation (ICCUBEA) (pp. 1-6). IEEE.
- [3] Lavee, G., Khan, L., & Thuraisingham, B. (2005, August). A framework for a video analysis tool for suspicious event detection. In Proceedings of the 6th international workshop on Multimedia data mining: mining integrated media and complex data (pp. 79-84).
- [4] Bedeli, M., Geradts, Z., & van Eijk, E. (2018). Clothing identification via deep learning: forensic applications. *Forensic sciences research*, 3(3), 219-229.
- [5] Chen, Q., Huang, J., Feris, R., Brown, L. M., Dong, J., & Yan, S. (2015). Deep domain adaptation for describing people based on fine-grained clothing attributes. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 5315-5324).
- [6] Zapotosky, M. (2021). FBI releases new footage of possible RNC, DNC pipe-bomb suspect, says person probably 'not from the area'. *The Washington Post*, NA-NA.
- [7] Narejo, S., Pandey, B., Esenarro Vargas, D., Rodriguez, C., & Anjum, M. R. (2021). Weapon detection using YOLO V3 for smart surveillance system. *Mathematical Problems in Engineering*, 2021, 1-9.
- [8] Grega, M., Matiołański, A., Guzik, P., & Leszczuk, M. (2016). Automated detection of firearms and knives in a CCTV image. *Sensors*, 16(1), 47.
- [9] Mehta, P., Kumar, A., & Bhattacharjee, S. (2020, July). Fire and gun violence based anomaly detection system using deep neural networks. In 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC) (pp. 199-204). IEEE.

- [10] Marbach, G., Loepfe, M., & Brupbacher, T. (2006). An image processing technique for fire detection in video images. *Fire safety journal*, 41(4), 285-289.
- [11] Dever, J., da Vitoria Lobo, N., & Shah, M. (2002, August). Automatic visual recognition of armed robbery. In *2002 International Conference on Pattern Recognition (Vol. 1, pp. 451-455)*. IEEE.
- [12] Yin, J. H., Velastin, S. A., & Davies, A. C. (1996). Image processing techniques for crowd density estimation using a reference image. In *Recent Developments in Computer Vision: Second Asian Conference on Computer Vision, ACCV'95 Singapore, December 5–8, 1995 Invited Session Papers 2 (pp. 489-498)*. Springer Berlin Heidelberg.
- [13] Ravanbakhsh, M., Nabi, M., Sangineto, E., Marcenaro, L., Regazzoni, C., & Sebe, N. (2017, September). Abnormal event detection in videos using generative adversarial nets. In *2017 IEEE international conference on image processing (ICIP) (pp. 1577-1581)*. IEEE.
- [14] Mehran, R., Oyama, A., & Shah, M. (2009, June). Abnormal crowd behavior detection using social force model. In *2009 IEEE conference on computer vision and pattern recognition (pp. 935-942)*. IEEE.
- [15] Heartexlabs. Heartexlabs/labelimg: LabelImg is now part of the label Studio Community. the popular image annotation tool created by Tzutalin is no longer actively being developed, but you can check out label studio, the open-source data labeling tool for images, text, hypertext, audio, video and time-series data.
- [16] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788)*.
- [17] Atik, M. E., Duran, Z., & ÖZGÜNLÜK, R. (2022). Comparison of YOLO versions for object detection from aerial images. *International Journal of Environment and Geoinformatics*, 9(2), 87-93.
- [18] Yang, F., Zhang, X., & Liu, B. (2022). Video object tracking based on YOLOv7 and DeepSORT. *arXiv preprint arXiv:2207.12202*.
- [19] Hussain, M., Al-Aqrabi, H., Munawar, M., Hill, R., & Alsoubi, T. (2022). Domain feature mapping with YOLOv7 for automated edge-based pallet racking inspections. *Sensors*, 22(18), 6927.
- [20] Wang, C. Y., Bochkovskiy, A., & Liao, H. Y. M. (2023). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 7464-7475)*.
- [21] Soudy, M., Afify, Y., & Badr, N. (2022). RepConv: A novel architecture for image scene classification on Intel scenes dataset. *International Journal of Intelligent Computing and Information Sciences*, 22(2), 63-73.
- [23] Zheng, J., Wu, H., Zhang, H., Wang, Z., & Xu, W. (2022). Insulator-defect detection algorithm based on improved YOLOv7. *Sensors*, 22(22), 8801.
- [24] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., & Torralba, A. (2016). Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2921-2929)*.