Real-time sign language to speech translation using CNNs

P. GDULA
ORCID: 0009-0003-3560-1252
piotr.gdula@student.wat.edu.pl

Military University of Technology, Faculty of Cybernetics
Kaliskiego St. 2, 00-908 Warsaw, Poland

This article presents the design, implementation, and empirical evaluation of an end-to-end software architecture for the real-time translation of the American Sign Language (ASL) alphabet into text and synthesized speech. The proposed system leverages the OpenCV library for video ingestion and the MediaPipe framework to extract 21 precise three-dimensional topological landmarks of the human hand, effectively isolating the gesture from environmental noise. A custom Convolutional Neural Network (CNN) built in TensorFlow and Keras processes these normalized skeletal representations to classify 26 static ASL characters. The model demonstrates a robust overall classification accuracy, performing exceptionally well on open hand topologies while highlighting occlusion-based challenges in complex gestures. Operating with exceedingly low latency, the system ensures fluid real-time human-computer interaction, providing a scalable and computationally efficient foundation for accessible communication tools without the need for specialized depth-sensing hardware.

Keywords: sign language translation, Convolutional Neural Networks, real-time processing.

DOI: 10.5604/01.3001.0055.9904

1. Introduction

Sign language constitutes the fundamental and indispensable medium of communication for the global deaf and hard of hearing communities. It operates as a highly sophisticated visual and spatial modality that fundamentally diverges from the acoustic propagation of spoken languages. This language relies on a complex multidimensional matrix of articulation wherein precise hand configurations, spatial trajectories, facial expressions, and dynamic body postures converge to convey nuanced semantic meaning. It is a persistent misconception among the public that sign languages are mere visual pantomimes or direct word for word manual translations of local phonic languages. They possess their own independent and highly evolved grammatical, syntactic, and structural rules that strictly govern linguistic expression. However, because most of the hearing population lacks even basic proficiency in this visual linguistic system, profound and pervasive communication barriers persist in everyday societal interactions. These barriers extend far beyond mere inconvenience. They frequently lead to systemic and often unintended social exclusion, severely limiting equitable access to educational opportunities and restricting the seamless integration of deaf individuals into competitive professional environments. Historically, attempts to bridge this communication chasm relied exclusively on human interpreters, which are constrained by availability, high financial costs, and a complete lack of privacy. In recent years, the rapid and exponential advancement of artificial intelligence, particularly the paradigm shifting developments in the fields of deep learning and computer vision [4, 5], presents an unprecedented technological opportunity to democratize accessibility. The deployment of advanced convolutional neural architectures to automatically analyze, interpret, and translate visual data captured from ubiquitous consumer grade RGB cameras strives to eliminate the traditional reliance on expensive sensory hardware or specialized data gloves. This approach establishes a highly scalable foundation for universal and automated sign language translation.

Within the broader holistic ecosystem of sign language communication, the practice of fingerspelling occupies a unique, critical, and irreplaceable niche. While natural sign languages primarily utilize specific dynamic gestures to represent entire words, abstract concepts, or complex ideas, fingerspelling acts as the vital bridge for out of vocabulary terms. The American Sign Language alphabet, for instance, consists of twenty-six distinct single handed static signs that correspond directly and sequentially to the standard Latin alphabet [1]. This manual alphabet

is predominantly utilized by the deaf community for the precise articulation of proper nouns, geographical locations, specific technical terminology, brand names, or obscure words that currently lack a dedicated and universally recognized ideographic gesture within the standard sign lexicon. Consequently, the ability of an automated system to accurately, robustly, and reliably translate the manual alphabet is not merely a supplementary feature. It serves as a mandatory and highly critical foundational step toward the overarching long term scientific goal of automating entire dynamic sign language sentence structures. Without the foundational capability to flawlessly interpret fingerspelled characters in strict real time, any comprehensive translation system would ultimately fail to capture the full semantic breadth of natural deaf communication, rendering it completely inadequate for complex real-world deployment.

The computational translation of this static fingerspelled signs into text and audible speech introduces significant and multifaceted technical challenges that must be rigorously addressed at both the algorithmic and software architectural levels. Gestures must be recognized with exceptional precision because the physiological differences between certain letters can be incredibly minute. Subtle occlusions, slight deviations in the angular orientation of phalanges, or the minor rotation of the wrist can drastically alter the predicted class and consequently change the intended semantic meaning of a spelled word. Furthermore, a highly viable production ready system must account for significant intra class variance. This means the algorithm must remain invariant to the subtle idiosyncratic differences in how individual users perform the exact same manual gesture based on their varying hand sizes, joint flexibility, or physical habits. This complexity is further exacerbated by the absolute necessity for extreme environmental resilience. The computer vision pipeline must maintain robust classification capabilities amidst unpredictable real-world noise, including suboptimal ambient illumination, extreme shadows, heterogeneous background clutter, and varying resolutions of standard web cameras [5, 14]. Taking these immense challenges into deep consideration, the primary objective of this research is to design, implement, and empirically evaluate a highly responsive end to end software architecture capable of translating the static American Sign Language alphabet into text and synthesized audio. By prioritizing extremely low computational latency and utilizing topological feature extraction to ensure environmental resilience, this study aims to deliver a practical and scalable translation tool that significantly mitigates the daily communication friction experienced by the deaf community [1, 3].

2. Related work and technological framework

Historically, automated sign language recognition systems have relied heavily on wearable hardware solutions such as data gloves embedded with flex sensors or specialized depth sensing cameras. While these specific hardware methods provide highly accurate spatial coordinates of the human hand, they impose significant physical constraints on the user, lack overall portability, and present high financial barriers that prevent widespread consumer adoption [13, 14]. As computational power increased over the last decade, modern vision-based approaches shifted toward utilizing standard and ubiquitous RGB cameras coupled with advanced machine learning algorithms [4, 5]. However, traditional computer vision techniques that process raw pixel data directly or rely on legacy feature matching algorithms like Haar Cascades or Histogram of Oriented Gradients exhibit severe structural vulnerabilities [6, 13]. These older systems are highly susceptible to variations in ambient illumination, disparate user skin tones, and complex background clutter. Consequently, their classification reliability degrades significantly in dynamic real world deployment scenarios where environmental consistency cannot be guaranteed [13, 14]. To build a truly accessible and robust translation tool, modern methodologies must pivot away from raw pixel analysis and embrace highly abstracted spatial feature extraction.

To isolate the hand gesture from ambient visual noise effectively without incurring massive computational costs, a robust geometric feature extraction mechanism is absolutely required. Deep learning frameworks optimized for generalized object detection are exceptional at bounding box localization but lack native architectures explicitly designed to articulate the intricate and multi jointed skeletal structure of a human hand [2]. Similarly, comprehensive pose estimation libraries demand extensive graphical processing power, rendering them largely unsuitable for strict real time deployment on standard consumer laptops or mobile hardware [2, 13]. To overcome these specific hardware and software limitations, the system proposed in this study utilizes the MediaPipe framework developed by Google [6, 7]. The MediaPipe Hand

Tracking pipeline is a highly optimized module capable of detecting and mapping twenty-one distinct three-dimensional landmarks on a human hand in strict real time [6, 7]. By extracting only, the precise spatial coordinates of this topological skeleton, the proposed system can discard all irrelevant background pixel data and variations in user appearance. This extraction process significantly reduces the dimensionality of the input feature space while flawlessly preserving the geometric integrity of the semantic sign [7, 8]. The utilization of such a lightweight topology generator is the fundamental technological pillar that allows the subsequent classification network to operate with extreme speed and accuracy.

The seamless execution of a complete end to end translation pipeline necessitates the strategic integration of several specialized software libraries alongside the primary landmark extraction tool [1]. The OpenCV library serves as the core engine for real time video ingestion, handling critical tasks such as camera frame capture, optimal image resizing, and preliminary matrix operations [8]. During the offline model development and training phase, the Scikit learn library was systematically employed to facilitate dataset splitting, data shuffling, and the categorical one hot encoding of alphabetical labels [2]. For the final human computer interaction modality, the system integrates a dedicated software library to synthesize the recognized text output into audible speech [1]. Selecting a local offline Text to Speech engine rather than relying on external cloud-based application programming interfaces eliminates network induced latency and potential internet connectivity failures. This specific architectural choice guarantees that the system maintains strict real time responsiveness, allowing for fluid and uninterrupted conversation processing directly on the local edge device [1, 3]. Through the synergistic integration of these highly optimized libraries, the translation architecture transcends the limitations of isolated algorithms and forms a cohesive and production ready communication platform.

3. Proposed system architecture

The architecture of the proposed sign language translation system is designed around a strictly modular and pipeline-oriented approach that executes across three distinct computational phases encompassing motion registration, robust data processing, and final output generation. The continuous workflow initiates the moment a standard computer webcam captures raw video frames of the user actively performing hand gestures. These incoming visual frames are immediately intercepted by the primary computer vision pipeline where the highly optimized MediaPipe framework thoroughly isolates the human hand and flawlessly extracts the exact spatial positions of twenty-one critical topological joints. To ensure that the subsequent classification network focuses exclusively on the structural topology of the gesture rather than irrelevant visual artifacts, the system entirely discards the original-colored pixel data along with any background interference. Instead of processing the raw image, the twenty-one extracted spatial coordinate pairs are plotted directly onto a blank white canvas to generate a clean and synthetic skeletal representation of the hand. This topological extraction process is crucial for maintaining high classification accuracy under widely varying environmental conditions because it eliminates the visual noise that traditionally confuses and degrades the performance of standard convolutional networks.

Once the skeletal representation is successfully generated, the system applies a series of rigorous preprocessing steps to meticulously prepare the visual data for deep learning inference. These synthetic skeletal images are mathematically standardized to a fixed spatial resolution of 128 by 128 pixels and subsequently converted into a single channel grayscale format to further reduce unnecessary computational overhead. The pixel intensities within these grayscale representations are then strictly normalized to fall within a numerical continuous range between zero and one. This specific data normalization process significantly accelerates the mathematical convergence of the neural network during the intensive training phase and substantially improves overall inference stability during real time execution. By providing the artificial neural network with a consistent and mathematically bounded input tensor, the translation system successfully mitigates the severe risk of exploding gradients and ensures that the internal activation functions operate within their most effective and predictable numerical domains.

The normalized grayscale skeletal representations are routed directly into a custom Convolutional Neural Network designed specifically for this complex translation task. Convolutional networks are uniquely engineered to process multidimensional visual data with a known grid like topology, making them exceptionally suited for advanced spatial image classification tasks. The core mathematical

assumption underlying this specific architectural choice is spatial locality, which dictates that the most critical geometric indicators such as the relative angles between adjacent fingers reside within clustered and neighboring pixel regions. The network comprehensively analyzes these localized regions by sequentially sliding learnable matrices known as filters or kernels across the entire input matrix. Mathematically, the discrete two-dimensional convolution operation for a given feature map G at specific coordinates (x, y) is formulated as follows:

$$\mathrm{G(x,y)} = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \mathrm{I(x+i, y+j)} * F(i,j) \quad (1)$$

where:
I – represents the input image matrix,
F – denotes the learnable filter weight matrix, and the variables m and n define the exact spatial dimensions of the applied kernel.

The operational mechanism of this discrete spatial filtering process across the input grid is illustrated in Figure 1 and Figure 2, showcasing the initial and terminal positions of the kernel sliding window.

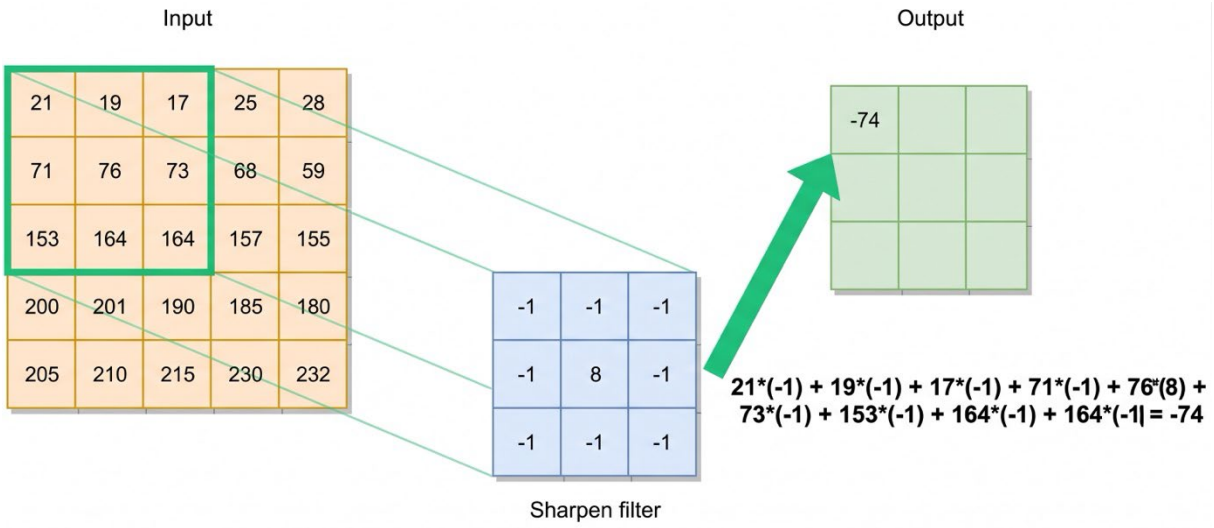


Fig. 1. First step of the 2D convolution operation with a 3 × 3 kernel applied to the input matrix.
Source: https://manduk.ai/pl/konwolucyjne-sieci-neuronowe-klasyfikacja-obrazow-czesc-2/

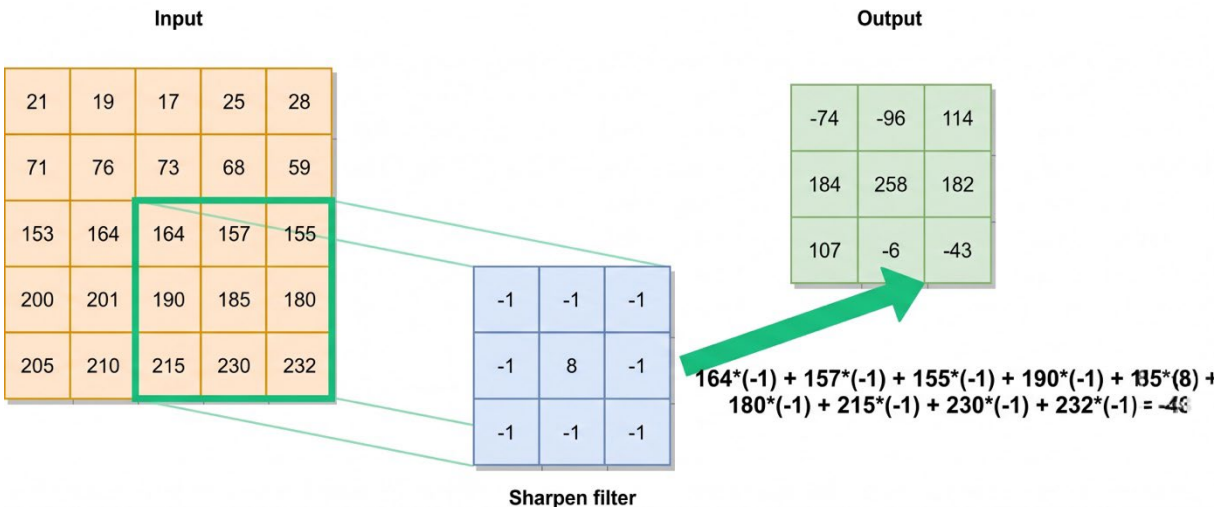


Fig. 2. Final step of the 2D convolution operation completing the output feature map.
Source: https://manduk.ai/pl/konwolucyjne-sieci-neuronowe-klasyfikacja-obrazow-czesc-2/

To introduce essential nonlinearity into the data processing pipeline, which ultimately allows the neural network to successfully learn and model highly complex nonlinear geometric configurations, the Rectified Linear Unit activation function is applied immediately following each individual convolution operation. Advanced down sampling layers, specifically max pooling mechanisms, are subsequently utilized to systematically reduce the spatial dimensions of the resulting feature maps, lower the overall computational burden on the processor, and grant the deep learning model a highly necessary degree of spatial translation invariance.

4. Convolutional Neural Network design and training

To train the image classification model to a high degree of mathematical precision and ensure its robustness in real world applications, a substantial and proprietary digital dataset was meticulously compiled. The complete dataset comprises twenty-six thousand synthetic skeletal images of human hands. These images are distributed with absolute equality across the twenty-six target classes of the American Sign Language alphabet, yielding exactly one thousand distinct training samples per individual character. This strict uniform distribution is critical to prevent any potential class imbalance bias during the training phase, which could otherwise skew the network towards more frequently represented gestures. Prior to feeding these visual samples into the artificial neural network, the standard alphabetical labels ranging from A to Z were mathematically converted into numerical integers and subsequently transformed into one hot encoded vectors. This specific vectorization technique allows the neural network to interpret the categorical target variables as distinct probability distributions rather than ordinal numerical values.

The custom convolutional architecture was constructed entirely within the TensorFlow and Keras computational environments [3, 9]. The model features a deep and hierarchical stack of processing layers specifically designed for aggressive and highly detailed spatial feature extraction. The network architecture begins with four consecutive convolutional blocks where the number of learnable filters progressively doubles across the sequential layers, moving systematically from sixty-four to one hundred twenty-eight, then two hundred fifty-six, and finally reaching five hundred twelve filters. All these convolutional operations utilize standardized three by three-pixel kernels. This progressive network depth allows the early computational layers to capture basic edge orientations and simple geometric lines while the deeper layers synthesize these elementary components into highly abstract representations of complete hand shapes. To ensure absolute mathematical stability during the gradient descent process and to completely prevent the occurrence of internal covariate shift, a Batch Normalization layer is integrated immediately following every single convolutional operation [12]. The spatial dimensions of the generated feature maps are systematically halved after each convolutional block using standard max pooling layers. Once the abstract spatial features are fully extracted by the convolutional base, the multidimensional matrices are flattened into a single dimensional vector and passed into a sequence of two fully connected dense layers containing five hundred twelve and two hundred fifty-six artificial neurons respectively. To aggressively combat the severe risk of overfitting, where the network simply memorizes the training data rather than generalizing geometric patterns, a high Dropout rate of fifty percent was strictly enforced on both dense layers [11]. This regularization technique randomly deactivates half of the active neurons during each forward pass of the training step. The terminal output layer consists of twenty-six neurons mapped to a Softmax activation function, which normalizes the raw mathematical network outputs into a clean probability distribution representing the likelihood of each individual American Sign Language character.

The model compilation process relied exclusively on the categorical cross entropy loss function, which accurately measures the performance of a classification model whose final output is a continuous probability value between zero and one. The crucial optimization of the network weights was achieved via the Adam algorithm [10]. This specific optimizer was selected because it utilizes adaptive learning rate properties to navigate the highly complex gradient descent landscape much more efficiently than traditional stochastic gradient descent methods. The overall training regimen was configured to run for a maximum of fifty complete epochs with a standardized processing batch size of sixty-four images per step. To further optimize computational time and hardware resources, an Early Stopping callback mechanism was implemented and remained continuously active throughout the learning phase. This intelligent mechanism automatically

terminated the training loop the exact moment the validation accuracy plateaued and stopped improving. This ensures the absolute preservation of optimal generalization capabilities without any unnecessary or redundant computational expenditure [2, 3].

5. Experimental Results

The classification efficacy and overall operational performance of the completed software translation system were subjected to rigorous empirical evaluation to comprehensively validate its theoretical capabilities. In terms of quantitative performance metrics calculated exclusively on the independent testing dataset, the trained convolutional neural network achieved a highly satisfactory and robust overall classification accuracy of 93%

A detailed mathematical extraction of specific performance metrics including precision, recall, and comprehensive statistical scores revealed that the computational model performs with near flawless accuracy when classifying alphabetical characters that exhibit highly distinct and visually open hand topologies. Characters characterized by wide finger separation and unobstructed visibility consistently yielded individual statistical scores approaching perfect confidence levels. This overwhelmingly positive outcome explicitly demonstrates that the artificial neural network can effortlessly differentiate basic geometric layouts where the human fingers are clearly separated and fully visible to the optical sensor. The strict utilization of standardized machine learning evaluation methodologies ensures that these empirical results are statistically significant and completely reproducible across varying validation subsets [2, 3].

To rigorously diagnose the class-specific classification boundaries and identify localized error distributions across all twenty-six target characters, a comprehensive statistical confusion matrix was algorithmically generated, as depicted in Figure 3.

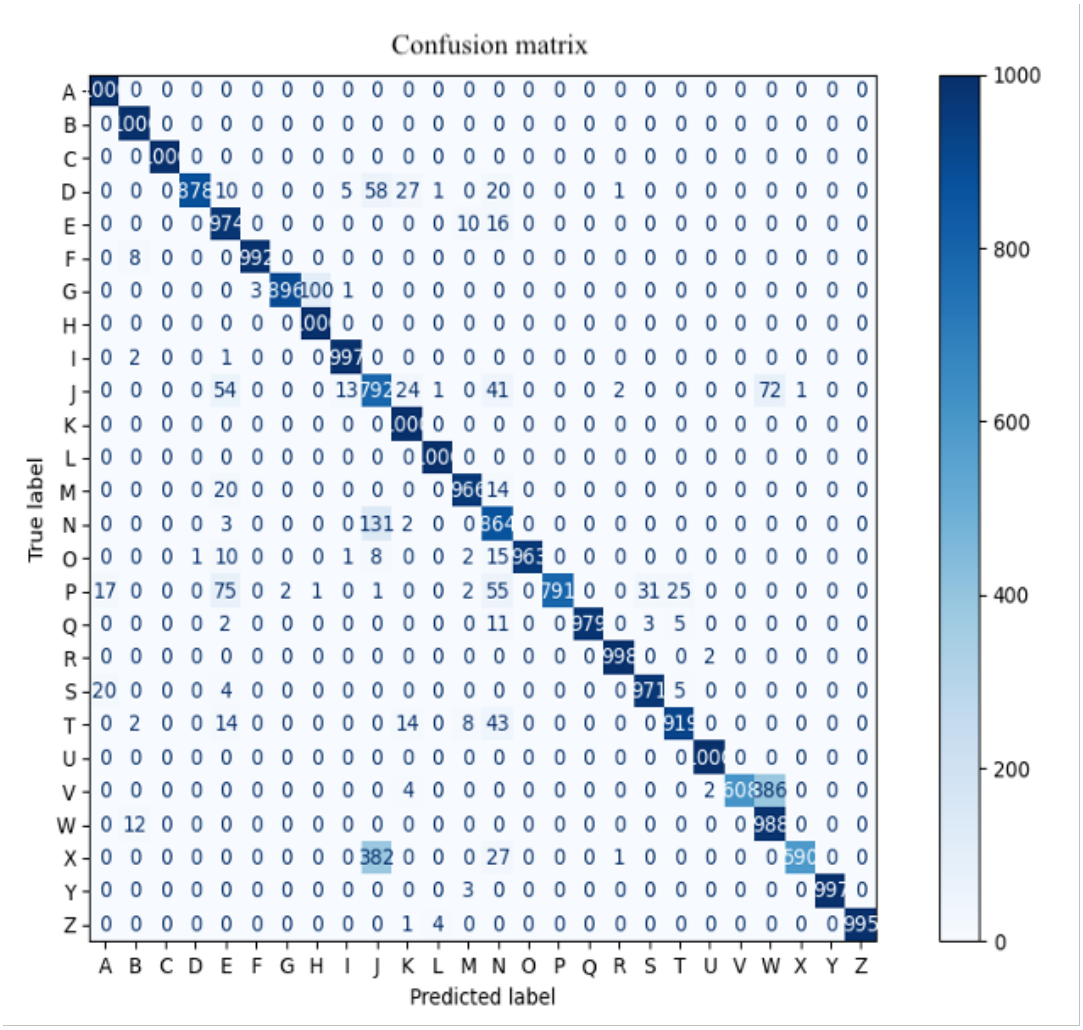


Fig. 3. Confusion matrix of the 26-class ASL alphabet classification evaluated on the independent test dataset
Source: own work

Conversely, the algorithmic analysis of the statistical confusion matrix brought to light specific and isolated classification bottlenecks that perfectly represent the inherent mathematical limitations of two-dimensional computer vision systems. The neural network demonstrated a noticeable and measurable drop in overall precision and recall when processing complex manual letters that require intricate overlapping of human digits. This localized performance degradation is directly tied to the physiological similarities of specific human gestures when they are forcefully compressed from a three--dimensional physical hand space onto a completely flat two-dimensional pixel plane [5]. Manual gestures that demand complex overlapping of human fingers or where the anatomical palm tilts sharply away from the central camera axis suffer from severe visual occlusion. In these highly specific instances, the underlying topological extraction framework occasionally struggles to accurately resolve spatial depth coordinates, leading to slightly distorted skeletal paths that ultimately confuse the downstream classification algorithm [13, 14].

Beyond raw statistical accuracy metrics, the practical viability and potential consumer adoption of any automated translation tool is heavily dictated by its system latency and overall operational throughput. Comprehensive empirical testing conducted across various standard hardware setups confirmed that the complete end to end processing time encompassing camera frame ingestion, spatial landmark tracking, synthetic skeleton rendering, network inference, and vocalized audio synthesis averaged at an exceptionally low duration of 180–200 ms. This specific latency profile falls well within the strict temporal limits required for fluid real time human interaction, creating a highly natural user experience where synthesized speech is generated almost simultaneously with the physical manual sign. Furthermore, to rigorously evaluate the long-term reliability of the interconnected software components, extensive stability tests were deliberately conducted over continuous operational windows of three hours. The integrated system exhibited zero measurable degradation in processing frame rates and absolutely no software memory leaks, confirming its structural integrity and absolute readiness for prolonged daily usage in demanding real-world environments.

6. Conclusion

In conclusion, this research successfully demonstrates that combining low-level topological feature extraction with a deeply structured convolutional neural network provides a powerful and highly efficient framework for real-time sign language translation. By intentionally discarding raw pixel values and background textures in favor of clean landmark skeletons, the system achieves remarkable immunity against environmental lighting fluctuations and domestic background clutter. Furthermore, because the input data is highly focused, the entire pipeline achieves rapid execution and high inference speeds using standard consumer-grade processing units, eliminating the need for expensive and specialized depth sensors [3].

The practical real-time classification performance and qualitative visual feedback of the developed system are illustrated in Figure 4.

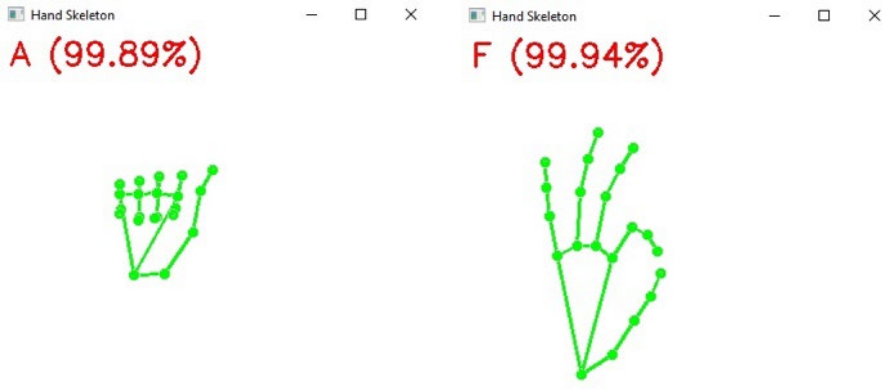


Fig. 4. Real-time classifications
Source: own work

As demonstrated by the empirical evaluation, the displayed probability confidence directly corresponds to the topological clarity of the presented sign. Gestures featuring fully extended digits consistently reach near-absolute certainty, whereas signs constrained by planar self-occlusions yield lower confidence values.

Despite these positive outcomes, the current iteration of the system possesses clear functional boundaries because the classification model is designed around static gesture analysis. It is fundamentally limited in its ability to interpret dynamic motion-based signs or complex temporal language structures.

Future development trajectories for these software components will focus on transitioning from isolated character recognition to continuous full-sentence translation by incorporating recurrent neural network architectures or modern transformer layers to analyze temporal sequences [3, 13]. Additionally, steps will be taken to adapt the framework to the unique grammatical structures of other regional sign languages, such as Polish Sign Language (PJM), and apply advanced model quantization techniques to enable native deployment onto mobile operating systems [3, 13].

7. Bibliography

[1] Goodfellow I., Bengio Y., Courville A., *Deep Learning*, MIT Press, Cambridge 2016.

[2] Géron A., *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*, O'Reilly Media, Sebastopol 2022.

[3] Chollet F., *Deep Learning with Python* (2nd Edition), Manning Publications, Shelter Island, 2021.

[4] LeCun Y., Bengio Y., Hinton G., "Deep learning", *Nature*, Vol. 521, No. 7553, 436–444 (2015).

[5] Szeliski R., *Computer Vision: Algorithms and Applications*, Springer, London 2022.

[6] Lugaresi C., et al., "MediaPipe: A Framework for Building Perception Pipelines", arXiv preprint arXiv:1906.08172 (2019).

[7] Zhang F., et al., "MediaPipe Hands: On-device Real-time Hand Tracking", arXiv preprint arXiv:2006.10214, (2020).

[8] Bradski G., "The OpenCV Library", *Dr. Dobb's Journal of Software Tools*, Vol. 25, No. 11, 120–123 (2000).

[9] Abadi M., et al., "TensorFlow: A System for Large-Scale Machine Learning", *12th USENIX Symposium on Operating Systems Design and Implementation* (*OSDI*), Savannah, USA,2016.

[10] Kingma D.P., Ba J., "Adam: A Method for Stochastic Optimization", *3rd International Conference on Learning Representations* (*ICLR*), San Diego, USA, 2015.

[11] Srivastava N., et al., "Dropout: A Simple Way to Prevent Neural Networks from Overfitting", *Journal of Machine Learning Research*, Vol. 15, No. 1, 1929–1958 (2014).

[12] Ioffe S., Szegedy C., "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift", *Proceedings of the 32nd International Conference on Machine Learning* (*ICML*), Vol. 37, 448–456, Lille, France, 2015.

[13] Oudah M., Al-Naji A., Chahl J., "Hand Gesture Recognition Based on Computer Vision: A Review of Techniques", *Journal of Imaging*, Vol. 6, No. 8, 73 (2020).

[14] Cheok M.J., Omar Z., Jaward M.H., "A review of hand gesture and sign language recognition techniques", *International Journal of Machine Learning and Cybernetics*, Vol. 10, No. 1, 131–153 (2019).

COMPUTER SCIENCE AND MATHEMATICAL MODELLING **23** 19–27 (2026)

Translacja języka migowego na mowę w czasie rzeczywistym z wykorzystaniem sieci CNN

P. GDULA

Artykuł przedstawia projekt, implementację oraz ewaluację empiryczną kompleksowej architektury oprogramowania do tłumaczenia alfabetu amerykańskiego języka migowego (ASL) na tekst i syntetyzowaną mowę w czasie rzeczywistym. Zaproponowany system wykorzystuje bibliotekę OpenCV do przetwarzania obrazu wideo oraz framework MediaPipe do ekstrakcji 21 precyzyjnych trójwymiarowych punktów topologicznych dłoni, co pozwala na skuteczną izolację gestu od szumu otoczenia. Autorska konwolucyjna sieć neuronowa (CNN), zbudowana w środowiskach TensorFlow i Keras, przetwarza znormalizowane reprezentacje szkieletowe w celu klasyfikacji 26 statycznych znaków ASL. Model wykazuje wysoką ogólną dokładność klasyfikacji, radząc sobie wyjątkowo dobrze z otwartymi układami dłoni, jednocześnie identyfikując wyzwania związane z okluzją w przypadku złożonych gestów. Dzięki wyjątkowo niskiemu opóźnieniu system gwarantuje płynną interakcję w czasie rzeczywistym, stanowiąc skalowalny i wydajny obliczeniowo fundament dla dostępnych narzędzi komunikacyjnych, które nie wymagają specjalistycznego sprzętu sensorycznego.

Słowa kluczowe: translacja języka migowego, konwolucyjne sieci neuronowe, przetwarzanie w czasie rzeczywistym.