Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X Volume 4, Issue 4, 2026

PARSIMONIOUS GESTURE BENCHMARKING FOR DUPLICATE-CONTAMINATED TOUCHLESS DOCUMENT INTERACTION

Basit Raza[*1], Samina Rajper[2], Noor Ahmed Shaikh[3], Zahid Hussain Shar[4], Iqra Hyder[5]

[*1,2,3,4,5]*Institute of Computer Science, Shah Abdul Latif University, Khairpur Mirs, Sindh, Pakistan*

[*1]basitraza.computerscience@gmail.com, [2]samina.rajper@salu.edu.pk, [3]noor.shaikh@salu.edu.pk, [4]0zahidhussain@gmail.com, [5]iqrahyder.cs42@gmail.com

DOI: https://doi.org/10.5281/zenodo.19690462

Keywords
Touchless Document Interaction; Static Hand Gesture Recognition; Leakage-Aware Benchmarking; Hash-Deduplicated Evaluation; Frugal Vision Models; Deployment-Oriented Gesture Interfaces

Article History
Received: 25 February 2026
Accepted: 04 April 2026
Published: 22 April 2026

Copyright @Author
Corresponding Author: *
Basit Raza

Abstract

Touchless document control is attractive for low-contact settings such as document browsing, command selection, and OCR triggering, yet small-vocabulary gesture interfaces are often reported without sufficient attention to benchmark hygiene or deployment cost. This study presents a leakage-aware, deployment-oriented benchmark analysis of a four-command static hand-gesture interface for touchless document interaction. We first audit the official split of a public benchmark and identify a serious evaluation issue 996 exact duplicate samples appear across the validation and test sets. To obtain a fairer assessment, we construct a hash-deduplicated clean split and compare two lightweight recognition routes a LandmarkMLP built on MediaPipe hand landmarks and normalized geometric features, and an image-based MobileNetV3-Small baseline trained on hand crops. On the clean split, MobileNetV3-Small achieves 99.90% accuracy and 0.9990 macro-F1 on the full test set, while LandmarkMLP reaches 99.48% accuracy and 0.9948 macro-F1 on samples with successful hand detection. Despite slightly lower recognition performance, LandmarkMLP is markedly more efficient, requiring only 0.505 ms average inference time and 0.289 MB of model storage, compared with 5.15 ms and 5.93 MB for the image baseline. Corruption experiments show strong performance under low light, blur, and JPEG compression, but also reveal that the landmark route's end-to-end robustness deteriorates under severe Gaussian noise because detector failures increase sharply. Overall, the results support the feasibility of low-cost touchless document interaction in controlled static-gesture settings, while emphasizing that fair evaluation and end-to-end reliability are as important as raw classification accuracy.

1. Introduction

Touchless interaction is especially useful in constrained settings where direct contact is inconvenient, undesirable, or operationally disruptive. In constrained document-centric workflows, users may need only a small set of reliable commands rather than a large gesture vocabulary for example, to browse a document, trigger OCR, confirm input, or stop an action. This design space is especially relevant to low-contact environments, kiosk-like systems, and lightweight hands-free interfaces, where usability depends not only on recognition accuracy but also

on simplicity, responsiveness, and deployment cost [1]-[9].

Prior work on hand-gesture recognition has established the importance of both geometric and appearance-based representations, and recent surveys show that the field has matured across recognition models, sensing modalities, and application domains [1]-[4]. At the same time, much of the literature continues to emphasize recognition performance alone. In practice, however, a usable touchless command interface must also be judged by latency, memory footprint, robustness under degraded visual conditions, and the reliability of any upstream detector on which the recognizer depends. This is particularly important for frugal systems, where the central question is not whether a large model can achieve high accuracy, but whether a compact pipeline can deliver stable behavior under realistic constraints [2], [4], [10].

A second issue is evaluation hygiene. Benchmark results can appear stronger than they really are when the underlying split protocol is not carefully audited. Broader computer vision research has long shown that dataset bias and hidden overlap can distort performance estimates and weaken claims of generalization [11], [12]. This concern is especially relevant for compact gesture benchmarks, where class vocabularies are small and repeated captures may unintentionally create optimistic validation or test behavior. Accordingly, fair benchmarking requires more than a train/test report; it requires explicit inspection of the split itself and a protocol that avoids contamination before model comparison begins.

This paper studies a four-command static hand-gesture problem for touchless document interaction under exactly that perspective. Rather than presenting a new recognition architecture, we ask a narrower and more practical question under a leakage-free protocol, how well can lightweight gesture-recognition pipelines support a low-cost touchless document interface? To answer this, we first audit the official split of a public four-class hand-gesture benchmark and find exact duplicate contamination between the validation and test partitions. We therefore construct a hash-deduplicated clean split and evaluate two lightweight routes under the same protocol a frugal landmark-based multilayer perceptron and a compact image-based MobileNetV3-Small baseline. The comparison is carried out using recognition metrics as well as deployment-relevant criteria, including latency, frames per second, model size, class-wise behavior, and corruption robustness.

This framing is intentionally scoped. The present study does not claim to solve unconstrained touchless interaction, nor does it treat a four-command static benchmark as a proxy for full human–computer interaction in the wild. Instead, it treats the task as a controlled feasibility setting in which three questions can be examined cleanly. First, does the benchmark remain highly learnable after duplicate-aware cleaning? Second, what trade-off emerges between a frugal landmark pipeline and a compact image route when both are evaluated fairly? Third, how should robustness claims be interpreted when classification accuracy depends on upstream hand detection?

Under the clean split, both routes achieve very strong recognition performance, confirming that the task is highly separable in its current controlled form. However, the two pipelines differ substantially in deployment profile. The image route attains the strongest full-sample recognition performance, whereas the landmark route offers a much smaller memory footprint and much lower inference latency. At the same time, the landmark pipeline exposes an important caveat corruption robustness cannot be interpreted only through post-detection classification metrics, because detector failure itself becomes part of the end-to-end system behavior. This distinction is central to fair reporting and is one of the main reasons why this paper treats benchmark design and evaluation protocol as first-class contributions rather than as auxiliary implementation details.

The contributions of this work are fivefold. First, we audit the benchmark and show that the official split contains exact duplicate contamination between validation and test data. Second, we construct a hash-deduplicated clean split for fair evaluation of a four-command touchless document-control task. Third, we compare a frugal landmark-based MLP with a compact image

baseline under accuracy, macro-F1, latency, FPS, model size, and corruption robustness. Fourth, we show that near-perfect recognition is achievable in this controlled setting, while the frugal landmark route offers a substantially stronger efficiency profile. Fifth, we identify an important caveat the corruption robustness of the landmark route must be interpreted jointly with hand-detector failure rates.

Overall, this work positions touchless document interaction as a practical, bounded systems problem rather than a broad AR/VR claim. Its main contribution is a leakage-aware and deployment-oriented benchmark study rather than a new recognition architecture.

Figure 1. System overview of the leakage-aware benchmark pipeline for the four-command touchless document interface

2. Related Work

2.1 Touchless document and command interfaces

Touchless interfaces have been studied in several low-contact or sterile-use scenarios, especially where physical interaction with a device is inconvenient or undesirable. Early work in medical environments showed that touchless image manipulation can reduce dependence on assistants while preserving operator sterility, motivating gesture-controlled access to visual content during procedures [5], [6]. Subsequent studies examined gesture preferences and usability in operating-room contexts, reporting that touchless control is most useful when the command set is small, interpretable, and matched to the workflow rather than designed for expressive richness alone [7], [8]. Beyond camera-based medical systems, newer hardware-oriented work has also explored display-integrated touchless interfaces, showing that hands-free input remains an active HCI topic even when the sensing mechanism differs from commodity RGB vision [9].

These studies are relevant to the present paper for two reasons. First, they reinforce that touchless control is often valuable not because it replaces all interaction, but because it supports a small number of high-value commands in constrained workflows. Second, they show that successful touchless systems are judged by usability, latency, and reliability, not by recognition accuracy alone. Our work adopts this narrower design philosophy. Instead of targeting broad free-form gesture input, we study a four-command static gesture interface for touchless document interaction, where a limited but dependable command vocabulary is a feature rather than a limitation.

2.2 Static hand-gesture recognition

Vision-based hand-gesture recognition has been surveyed extensively, both from the broader HCI perspective and from the viewpoint of modern deep-learning pipelines [1]-[4]. These reviews show a common methodological divide between appearance-based models, which learn directly from images, and structure-based models, which operate on hand pose, landmarks, or skeletal

abstractions. Appearance-based methods often achieve strong recognition performance but can be more sensitive to visual nuisance factors and are usually heavier in compute. Landmark- or skeleton-based approaches, by contrast, can be much more compact and interpretable, although their end-to-end performance depends on the quality of the upstream detector [2]-[4].

For compact image-based recognition, mobile-oriented convolutional networks remain especially relevant. MobileNetV3 is one of the strongest representatives of this design philosophy, offering a carefully engineered balance among accuracy, latency, and memory for on-device use [10]. This makes it a natural baseline when the objective is not only to classify well, but also to assess whether a recognizer is viable in low-cost or edge-style deployment settings. In contrast, lightweight landmark pipelines can reduce input dimensionality dramatically by replacing raw appearance with normalized hand geometry. Such representations are especially attractive for small-vocabulary command tasks, where the discriminative burden is lower and the benefits of frugality are more pronounced.

The present work sits at this junction. Rather than introducing a novel backbone, it compares two established routes that reflect different deployment philosophies a frugal geometric recognizer and a compact appearance-based CNN. This comparison is intentionally task-specific. Our aim is not to argue that one representation class universally dominates the other, but to understand how the trade-off behaves in a constrained four-command document-control setting.

2.3 Frugal and deployment-aware vision systems

The growing importance of on-device and resource-constrained inference has increased interest in models that are not only accurate but also small, fast, and easy to deploy. In gesture-recognition research, this is particularly important because many proposed systems are evaluated in desktop or laboratory settings even though their target use cases often imply embedded, mobile, or low-cost operation [2], [4]. Compact backbones such as MobileNetV3 exemplify the move toward hardware-aware design, where network architecture is shaped by latency and memory constraints rather than by accuracy alone [10].

A deployment-aware perspective is especially valuable for touchless command interfaces. In such settings, model size affects portability, latency affects perceived responsiveness, and detector reliability affects end-to-end usability. For a four-command interface, the relevant question is not merely whether recognition is possible, but whether a pipeline can achieve a favorable accuracy–latency–memory trade-off under simple hardware assumptions. This is where frugal geometric pipelines become particularly interesting they may sacrifice some appearance detail, but they can offer compelling practical advantages when inference cost matters more than squeezing out the last fraction of a percentage point in accuracy.

Our study therefore treats deployment metrics as first-class outcomes rather than supplementary engineering details. The comparison between the landmark-based MLP and MobileNetV3-Small is motivated precisely by this trade-off one route asks how far geometric compression can go in a small command space, while the other asks what accuracy can be retained by a compact image baseline designed for mobile-scale inference [10].

2.4 Evaluation leakage and benchmark hygiene

Benchmark validity depends not only on model design but also on the integrity of the evaluation protocol. More generally, computer vision literature has repeatedly shown that dataset bias, hidden overlap, and poorly specified splits can inflate confidence and lead to misleading claims of generalization [11], [12]. In parallel, data stewardship literature has emphasized that reproducible evaluation depends on transparent metadata, clear reporting, and reusable artifacts rather than on raw files alone [11]. These concerns are directly relevant to gesture recognition, especially in compact datasets where repeated captures, narrow backgrounds, or uncontrolled duplication can make a benchmark appear harder—or easier—than it truly is.

This issue becomes even more important when the task is framed as a system benchmark rather than

a pure classifier comparison. Recent illumination and color-vision benchmark work has increasingly emphasized explicit capture-factor reporting, machine-readable metadata, and reproducible summary assets [13]–[16]. Although those datasets target different visual problems, they offer a useful methodological lesson benchmark usefulness increases when the data release makes potential sources of variation—and potential sources of bias—visible and auditable. The recent SCD release is an example of this broader reporting philosophy, pairing image data with structured metadata, summary statistics, and reproducible figure generation [16].

A related methodological lesson comes from outside the gesture domain. In recent filtered-retrieval work, correctness-safe evaluation has been framed as the requirement that each method be judged within the same valid candidate set, rather than against a mismatched or overly permissive ground truth [17]. That paper is not about gesture recognition, but its evaluation principle is directly relevant here a benchmark should be structured so that the reported score genuinely corresponds to the decision space in which the model operates. In our setting, this means that duplicate contamination must be removed before performance is interpreted, and that post-detection classification quality must be distinguished from end-to-end interface reliability. Accordingly, our work places unusual emphasis on split auditing, duplicate-aware cleaning, and deployment-aware reporting. Unlike prior work centered mainly on model innovation, our focus is on fair evaluation and practical model trade-offs for a constrained command interface.

3. Task Definition and Benchmark Audit

3.1 Four-Command Interface Scenario

We study a constrained static hand-gesture task for touchless document interaction. The command vocabulary consists of four dataset-aligned classes delete**,** input**,** send_to_OCR, and stop. These labels are kept exactly as defined in the dataset and are used consistently throughout training, evaluation, and error analysis. We intentionally avoid remapping them into broader semantic names such as open, next, or close, because doing so would introduce an unnecessary mismatch between the benchmark labels and the actual experiments. Under this formulation, the task is not intended to represent open-ended gesture interaction; rather, it serves as a compact command interface in which a small vocabulary allows fair comparison between lightweight recognition routes.

3.2 Dataset Structure

The benchmark contains four gesture classes organized in the official directory structure of training_set, valid_set, and test_set. In the released split, each class appears with the same count inside each partition 2,099 images per class in the training set, 249 per class in the validation set, and 349 per class in the test set. This corresponds to 8,396 training images, 996 validation images, and 1,396 test images, for a total of 10,788 images. At first glance, this layout appears balanced and suitable for direct benchmarking. However, a split-level audit shows that balance alone is not sufficient to guarantee fair evaluation.

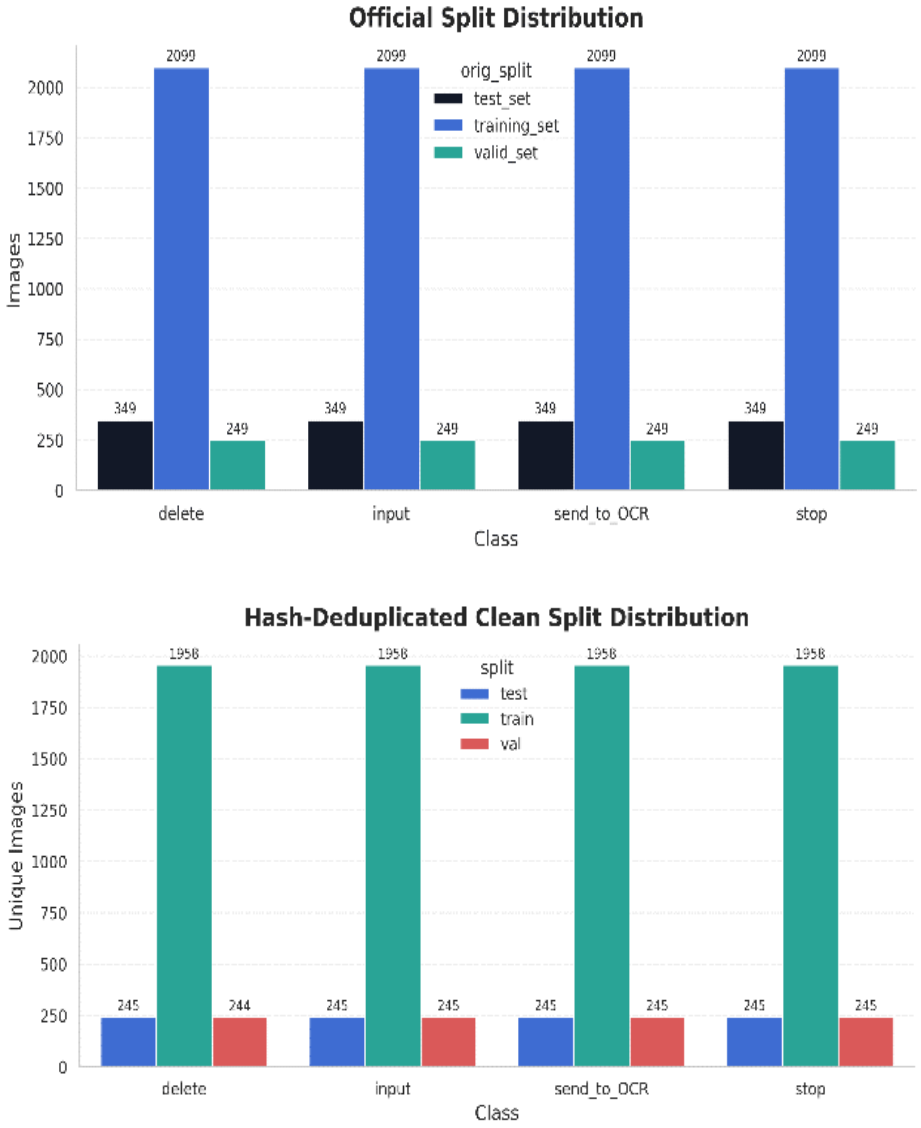


Figure 2. Class distributions before and after duplicate-aware cleaning. (a) Official split distribution. (b) Hash-deduplicated clean split used for all main experiments.

3.3 Duplicate Leakage Analysis

A hash-based duplicate audit reveals that the official protocol is contaminated. There is no exact duplicate overlap between training and validation, and no exact duplicate overlap between training and test. In contrast, the validation and test sets share 996 exact duplicates. Since the validation set itself contains 996 images, this means that the official validation partition is fully duplicated inside the test partition. As a result, the official split cannot be treated as a reliable basis for model selection and final reporting, because it risks optimistic assessment of generalization. This issue is not a minor bookkeeping artifact; it directly affects the credibility of any benchmark result reported on the original protocol.

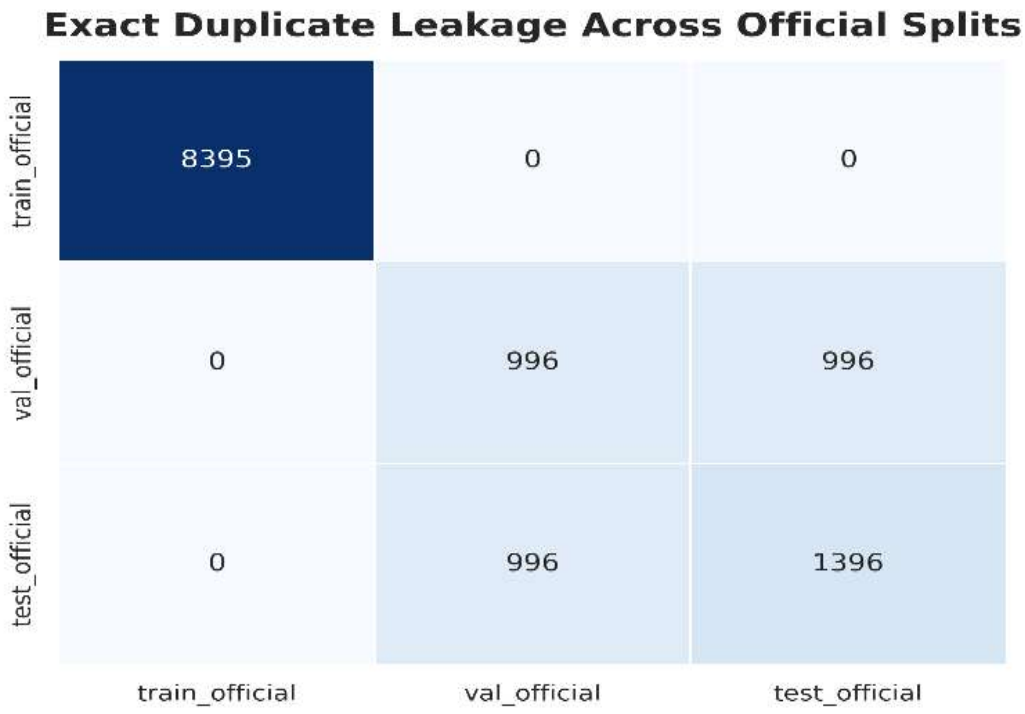


Figure 3. Exact duplicate leakage across the official benchmark split. The validation and test partitions share 996 exact duplicate images, showing that the original protocol is contaminated

3.4 Clean Hash-Deduplicated Split

To obtain a fairer evaluation setting, we construct a clean hash-deduplicated split. Exact duplicate files are identified by file hash, and only one instance of each unique image is retained before re-splitting. The resulting unique set is then divided using a stratified clean split so that class balance is preserved across partitions. This process yields 1,958 training images per class, 244-245 validation images per class, and 245 test images per class. The final clean benchmark therefore contains a balanced and leakage-free evaluation protocol while preserving the original four-class task definition. This duplicate-aware protocol is necessary to prevent optimistic assessment and to support fair comparison across gesture-recognition routes.

4. Methodology

4.1 Overview and Problem Formulation

To make the benchmark definition explicit, we formulate the task at the dataset, split, and prediction levels before describing the two recognition routes in detail. Let the benchmark consist of image-label pairs drawn from four command classes delete, input, send_to_OCR, and stop, together with their official split identities. Since the main goal of this study is fair evaluation under possible split contamination, the formulation includes both exact hash-based duplicate auditing and clean split construction before model training. We then define two lightweight prediction routes a detect-then-classify landmark pipeline and a compact image-based baseline. The key equations summarize the benchmark variables, duplicate condition, clean-set definition, route-specific prediction functions, and main recognition metrics, while Algorithm 1 outlines the full leakage-aware evaluation pipeline used in this work.

(1) $\mathcal{D} = \{(x_i, y_i, s_i)\}_{i=1}^{N}, \quad y_i \in \mathcal{Y}$

Dataset with image x_i, label y_i, and official split identity s_i.

(2) $\mathcal{Y} = \{\text{delete, input, send\_to\_OCR, stop}\}$

Four-command label set used throughout the benchmark.

(3) $h(x_i) = h(x_j),\ s_i \neq s_j \Rightarrow$ duplicate leakage

An exact file-hash match across different official partitions indicates contamination.

(4) $\mathcal{D}_u = \{(x_i, y_i) \in \mathcal{D} \mid h(x_i)$ retained once after deduplication$\}$

Unique image set used to build the clean evaluation protocol.

(5) $\hat{y}_{\text{LMK}} = f_{\text{MLP}}(\phi(g(x))), \quad \hat{y}_{\text{IMG}} = f_{\text{CNN}}(c(x))$

Landmark route and image route prediction functions.

(6) $\text{Accuracy} = \frac{1}{M}\sum_{i=1}^{M}\mathbf{1}(\hat{y}_i = y_i), \quad \text{Macro-F1} = \frac{1}{|\mathcal{Y}|}\sum_{k=1}^{|\mathcal{Y}|}\text{F1}_k$

Main recognition metrics used for comparison.

Figure 4. Problem formulation and key equations for duplicate-aware four-command gesture benchmarking

Merge all official partitions into $\mathcal{D}^{\text{off}}$
for each image $x_i \in \mathcal{D}^{\text{off}}$ **do**
    Compute exact file hash $h(x_i)$
end for
Audit duplicate overlap across official partitions using hash matches
Remove repeated hashes to obtain the unique set $\mathcal{D}_u$
Construct a stratified clean split:

$$\mathcal{D}_u \rightarrow \mathcal{D}_{train},\ \mathcal{D}_{val},\ \mathcal{D}_{test}$$

for each image $x_i \in \mathcal{D}_{train} \cup \mathcal{D}_{val} \cup \mathcal{D}_{test}$ **do**
    Run hand detection using MediaPipe
    if hand is detected **then**
        Extract landmarks and compute normalized geometric features
        Save hand crop from detected bounding box
    else
        Mark landmark route as detection failure
        Use full image as fallback for image route
    end if
end for
Train LandmarkMLP on successfully detected training samples
Train MobileNetV3-Small on clean split images/crops
Evaluate both routes using accuracy, macro-F1, weighted-F1, AUROC, latency, FPS, model size, and robustness
return leakage audit results, clean-split recognition results, and deployment-oriented comparison

Figure 5. Leakage-aware benchmark pipeline used for clean split construction, dual-route training, and deployment-oriented evaluation

4.2 Landmark Route

The first recognition route is a detect-then-classify pipeline built for frugal inference. Each image is first processed with MediaPipe Hands in single-hand static-image mode to localize one hand and estimate its 21 landmarks. To reduce avoidable detector failures, landmark extraction uses a small fallback sequence the original RGB image is attempted first, followed by contrast-equalized and slightly brightened versions when necessary. If detection succeeds, the landmark coordinates are converted into a normalized geometric representation. Specifically, the hand is translated to the wrist origin, rotationally aligned using the wrist-to-middle-finger direction, and scale-normalized by the maximum radial extent of the hand. From this normalized pose, we derive a compact feature vector consisting of flattened landmark coordinates, wrist-to-joint distances, bone-length descriptors, and a set of inter-joint angular features. These descriptors are then classified by a lightweight multilayer perceptron (MLP) with batch normalization, GELU activations, and dropout regularization. This route is intentionally designed to test how far a compact geometric representation can support a small-vocabulary touchless interface under tight efficiency constraints.

4.3 Image Route

The second route is a compact appearance-based baseline. When hand landmarks are available, the detected hand region is converted into a bounding box and expanded with a small safety margin to preserve local context. The cropped hand image is then resized and passed to MobileNetV3-Small, which serves as the image-based recognizer. If no valid crop can be formed, the full image is used as a fallback so that the image route remains evaluable on the complete split. Standard lightweight augmentation is applied during training, including modest rotation, horizontal flipping, and color jitter, while inference uses a deterministic resize-and-normalize pipeline. This branch provides a strong compact CNN baseline for appearance-driven recognition without moving to large or computationally expensive architectures.

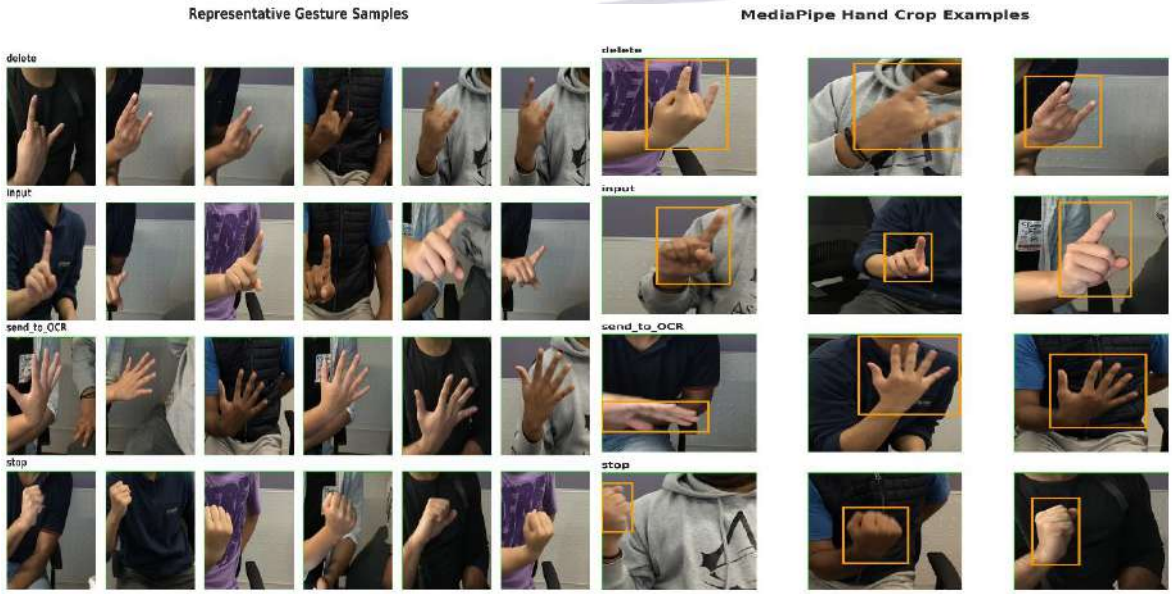


Figure 6. Visual examples from the benchmark and the crop-generation pipeline. (a) Representative gesture samples across the four classes. (b) Example hand crops produced for the image-based route

4.4 Rationale for the Two-Route Design

The two routes are selected to represent complementary deployment philosophies. The landmark pipeline is a frugal geometric route it compresses the input into a small hand-structure descriptor and is expected to provide very low latency and memory cost, albeit at the price of dependency on upstream hand detection. The image route is a **c**ompact appearance-based baseline it retains richer visual information and therefore serves as a stronger reference for recognition quality on the full sample set. Evaluating both routes under the same clean split enables a meaningful efficiency-versus-accuracy comparison. This comparison is central to the paper because the main question is not whether gesture classification is possible in principle, but how different lightweight pipelines behave when fairness, robustness, and deployability are considered together.

4.5 Evaluation Metrics

We evaluate the two routes using both recognition and deployment-oriented metrics. Recognition quality is reported through accuracy, macro-F1, weighted-F1, and macro AUROC under one-vs-rest multiclass evaluation. Efficiency is characterized using mean inference latency, FPS, and model size. Reliability is further analyzed through per-class performance, confusion matrices, hand-detection rate, and corruption robustness under low light, high brightness, blur, Gaussian noise, and JPEG compression. For the landmark route, classification performance is reported on successfully detected hands, while detection statistics are analyzed separately. This distinction is important because the landmark pipeline's end-to-end behavior depends not only on the MLP classifier but also on the stability of the upstream hand detector under visual degradation.

5. Experimental Protocol

5.1 Training Setup

All main experiments are conducted on the clean hash-deduplicated split, and all headline results in this paper are based on that protocol rather than the contaminated official split. The LandmarkMLP is trained with AdamW using a learning rate of 3×10^{-3} for up to 50 epochs. The MobileNetV3-Small baseline is also trained with AdamW, using a learning rate of 3×10^{-4} for up to 18 epochs. In both cases, training uses cosine learning-rate decay, label smoothing, and early stopping with a patience of 8 epochs based on validation macro-F1. For the image route, lightweight augmentation is applied during training, including small random rotations, horizontal flips, and moderate color jitter. This setup keeps the comparison practical and stable while avoiding unnecessary training complexity.

5.2 Fair Comparison Protocol

The two routes are compared under the same clean split, but their evaluation scopes are not identical and must be interpreted carefully. The image baseline is tested on the full split, because it can fall back to the full image even when a precise hand crop is unavailable. In contrast, the landmark classifier is evaluated only on samples with successful hand detection, since landmark features do not exist when detection fails. For this reason, detector failure rates are not treated as a minor preprocessing detail; they are part of the overall system analysis. This protocol allows us to compare recognition quality and efficiency fairly while keeping the role of detection reliability explicit.

5.3 Perturbation Study

To test robustness beyond the clean test set, we apply a controlled perturbation study on a held-out test subset. Five corruption types are used low light, high brightness, Gaussian blur, Gaussian noise, and JPEG compression. These perturbations are chosen because they reflect common degradations in practical camera-based interaction. For the landmark route, hand detection is repeated on each perturbed image before classification, so the reported behavior captures both classifier sensitivity and detector stability. For the image route, the recognizer is evaluated directly on the perturbed crop or fallback full image. This design helps separate "accurate when detected" behavior from true end-to-end robustness.

5.4 Interface-Level Smoothing

In addition to frame-level recognition, we include a small exploratory smoothing analysis to mimic a simple command-confirmation mechanism in a pseudo-stream setting. A short temporal voting filter is applied over successive predictions, using a confidence threshold and a stability rule before a command is triggered. This analysis is not presented as a core contribution or as a full interaction study. Its purpose is only to provide an initial indication of how conservative temporal filtering may affect trigger accuracy and coverage in a lightweight interface setting.

6. Results

6.1 Benchmark Audit Results

We begin with the benchmark audit because it directly affects the validity of all later results. The official split is contaminated there is no exact duplicate overlap between train and validation, and no exact duplicate overlap between train and test but the validation and test partitions share 996 exact duplicate images. This means the original protocol cannot be treated as a reliable basis for model selection and final reporting. For this reason all main claims in this paper are based on the clean hash-deduplicated split, not on the official split.

6.2 Recognition Results on the Clean Split

Under the clean protocol, both lightweight routes achieve very high recognition performance. LandmarkMLP reaches 0.9948 accuracy and 0.9948 macro-F1, while MobileNetV3-Small reaches 0.9990 accuracy and 0.9990 macro-F1. These results show that the cleaned benchmark remains highly separable even after duplicate removal. This makes the task useful not as a test of architectural novelty, but as a fair setting for comparing a frugal geometric pipeline against a compact appearance-based baseline.

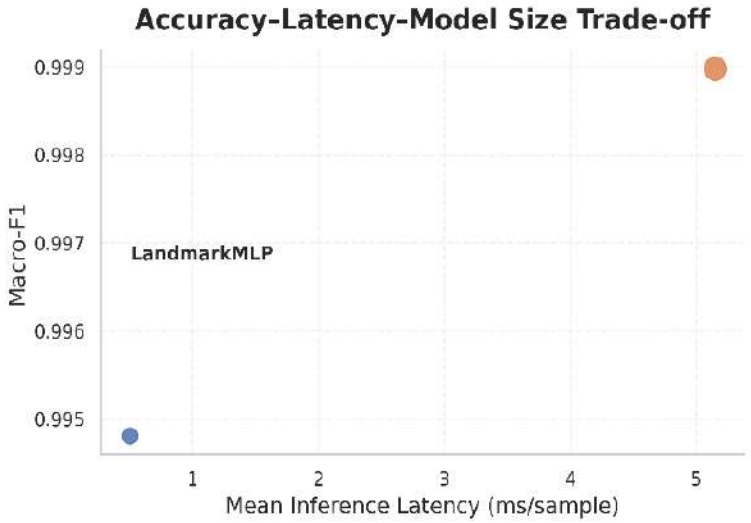


Figure 7. Accuracy–latency–model size trade-off between the two lightweight routes. MobileNetV3-Small gives the strongest recognition accuracy, while LandmarkMLP provides a much stronger efficiency profile

6.3 Efficiency Trade-Off

The efficiency comparison is one of the clearest outcomes of this study. LandmarkMLP requires only 0.505 ms mean inference time, achieves about 1980 FPS, and occupies just 0.289 MB. In contrast, MobileNetV3-Small requires 5.15 ms, reaches about 194 FPS, and uses 5.93 MB. Therefore, the image model delivers the strongest recognition accuracy, but the landmark model offers a much stronger deployment profile in latency, throughput, and storage. This makes the

landmark route especially attractive for low-cost and edge-style settings where compactness matters.

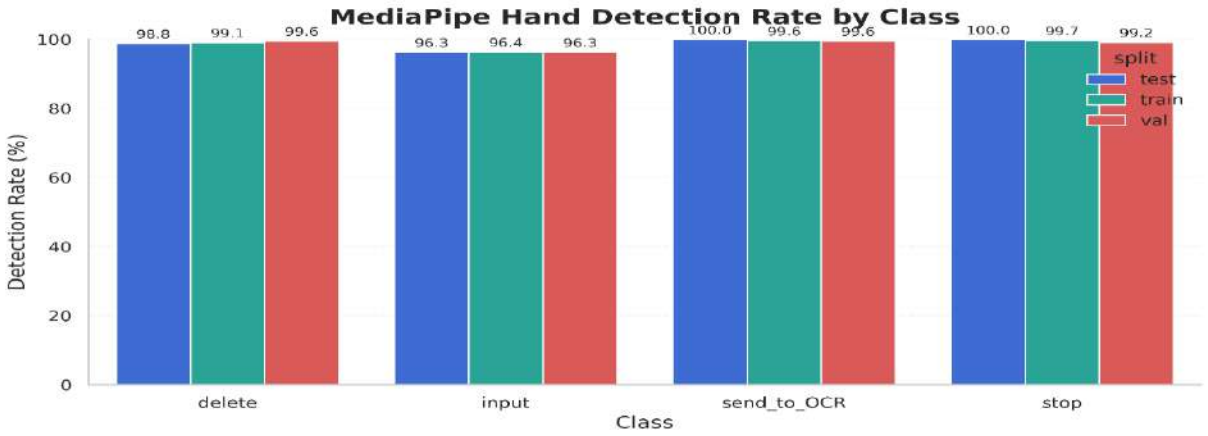


Figure 8. MediaPipe hand-detection rate by class on the clean split. Detection remains high overall, although the input class is consistently the weakest-detected category

6.4 Detection Reliability and Class-Specific Behavior

Hand detection is generally reliable across the clean split, which supports the practical use of the landmark route. However, detection quality is not identical across classes. The input gesture is the weakest-detected class, with a detection rate of about 96.3%, while the other classes remain close to or above 99% in most cases. Class-wise F1 scores are consistently high for both routes, indicating that the task is not dominated by one easy class alone. At the same time, the slightly lower detection rate for input adds an important realism point the landmark route depends not only on the classifier, but also on the stability of the upstream detector.

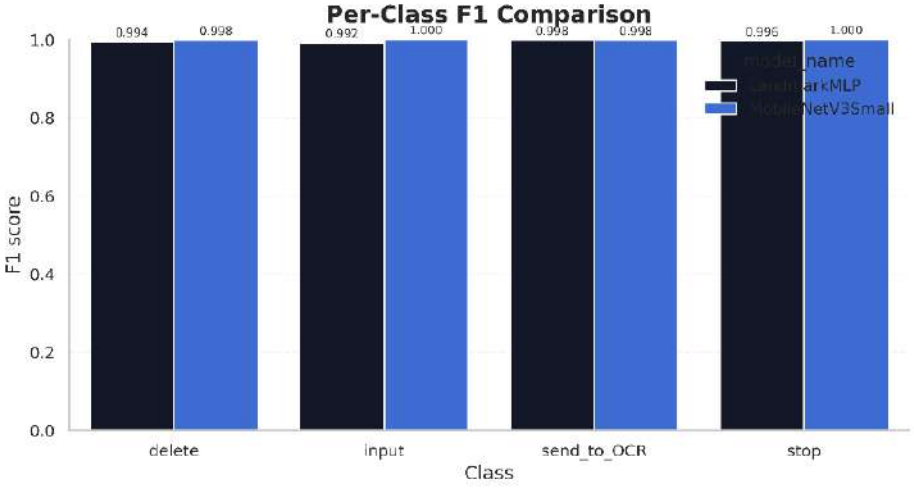


Figure 9. Per class F1 comparison on the clean split. Both routes remain strong across all four commands, with only small class-level differences

6.5 Corruption Robustness

The corruption study shows that both routes remain strong under low light, high brightness, blur, and JPEG compression. In other words, moderate visual degradation does not substantially change the main conclusion of the paper.

However, Gaussian noise exposes different failure modes. The image route shows a noticeable drop in recognition quality, while the landmark route suffers from a different problem detector instability. Under strong Gaussian noise, LandmarkMLP still reaches about 0.8977 macro-F1 on successfully detected hands, but the hand detector fails on about 44% of corrupted samples. Therefore, the robustness of the landmark route must be interpreted jointly with detector failure, not only through post-detection classification scores.

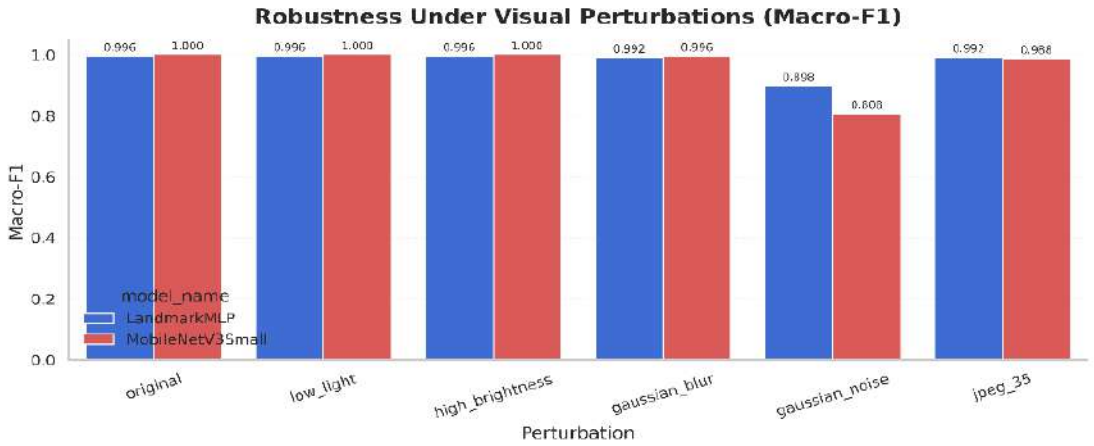
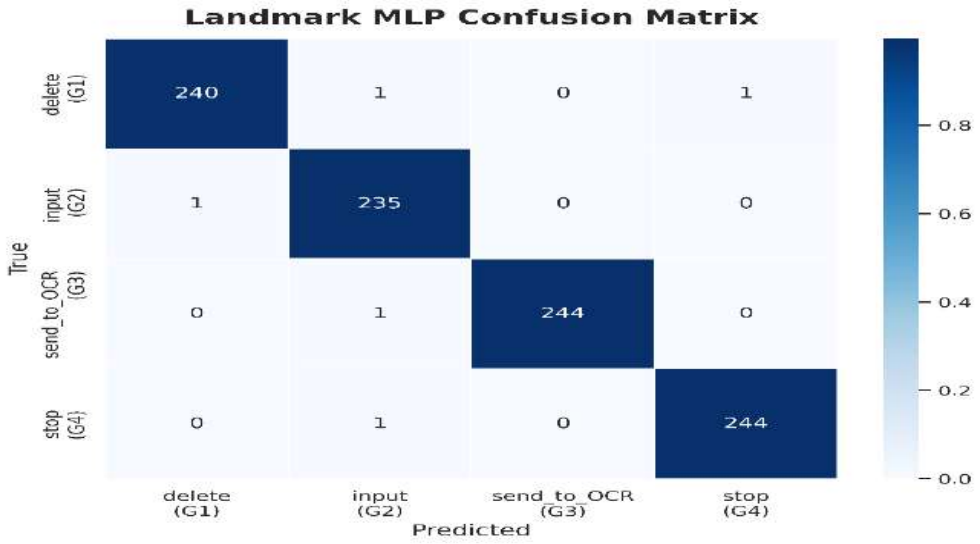


Figure 10. Macro-F1 under visual perturbations. Both routes remain strong under low light, brightness change, blur, and JPEG compression, while strong Gaussian noise reveals route-specific failure behavior

6.6 Confusion Analysis

Both routes show minimal confusion on the clean test set. The remaining errors are few and appear isolated rather than systematic. This result is consistent with the overall pattern of near-perfect performance and suggests that, within this controlled benchmark, the four commands are visually well separated after duplicate-aware cleaning.



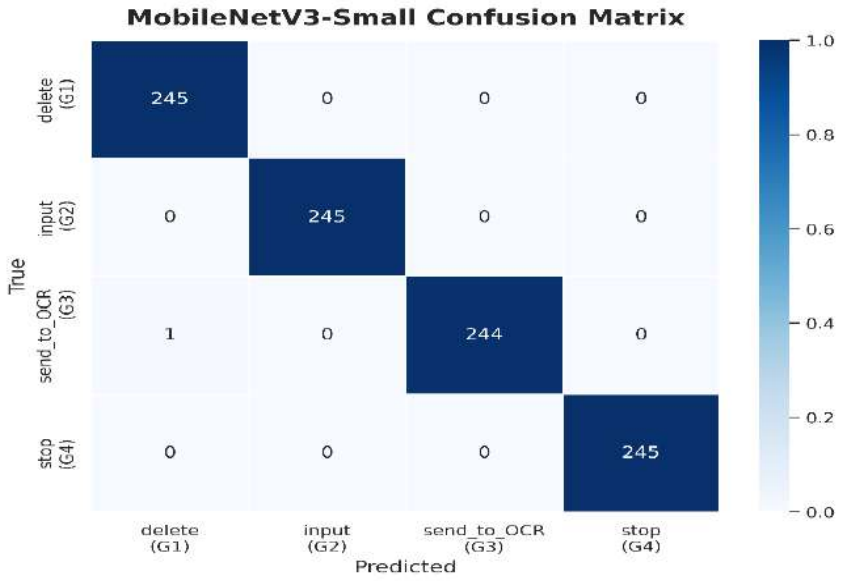


Figure 11. Confusion matrices on the clean test split. (a) LandmarkMLP. (b) MobileNetV3-Small. Both routes show minimal confusion, with only a few isolated errors

6.7 Exploratory Interface-Level Smoothing

We also performed a small exploratory smoothing analysis to mimic a conservative command-confirmation rule over a pseudo-stream of predictions. The raw prediction accuracy remains high, but smoothing produces only modest trigger accuracy and very low trigger coverage. In our current setup, raw accuracy is about 0.988, smoothed trigger accuracy is about 0.889, and trigger coverage is only about 0.036. This suggests that the present smoothing rule is too conservative for practical use. We therefore treat this result as preliminary and do not consider it a central contribution of the paper.

7. Discussion

After duplicate removal, the benchmark becomes far more trustworthy as an evaluation setting. This matters because the original split could have produced overly optimistic conclusions, especially when validation and test data overlap exactly. Under the clean protocol, both models still achieve extremely strong results, which shows that the task remains highly separable even after contamination is removed. For this reason, the main value of this paper is not to claim state-of-the-art recognition, but to provide a fairer benchmark and a clearer view of the trade-off between recognition quality and deployment efficiency.

The comparison between the two routes is also informative. The image model gives the strongest full-sample recognition baseline, making it the safer choice when the goal is maximum classification performance on all available inputs. In contrast, the landmark route is especially appealing when very small model size and very low latency are important. Its efficiency profile is strong enough to make it a realistic option for low-cost and edge-oriented settings, but its practical behavior cannot be judged from classifier scores alone because it depends on stable hand detection. This point is especially important for deployment. In a real touchless interface, end-to-end behavior matters more than conditional accuracy after detection succeeds. A model that is highly accurate on detected samples may still become unreliable if the upstream detector fails under noise or other visual degradation. Therefore, the present results should be read as evidence of feasibility in a controlled static-gesture setting, not as proof of broad real-world readiness. The present results support feasibility, but they do not yet justify broad claims about unconstrained real-world touchless interaction.

8. Limitations

This study has several limitations. First, the task includes only four static commands, so the findings should not be extended to larger gesture vocabularies, dynamic gestures, or continuous interaction scenarios. Second, the benchmark appears visually controlled, with limited variation in background, viewpoint, and capture conditions, which likely makes the task easier than less constrained real-world settings. Third, the landmark route is evaluated conditionally on successful hand detection, so classifier performance and end-to-end interface reliability must be interpreted separately. Fourth, the current study does not include a user study, interaction-time analysis, fatigue assessment, or cross-device evaluation, so usability and deployment behavior remain only partially examined. Finally, all results come from a **single benchmark source**, and broader generalization to different users, environments, and capture pipelines has not yet been established.

9. Conclusion

This paper presented a leakage-aware benchmark study of a four-command touchless document interface using two lightweight gesture-recognition routes. Our audit showed that the official split is contaminated by exact duplicates between validation and test data, which makes duplicate-aware evaluation necessary. To address this, we introduced a clean hash-deduplicated split and used it as the basis for all main results.

Under this fairer protocol, the benchmark remained highly separable, and both routes achieved very strong recognition performance. MobileNetV3-Small produced the strongest overall recognition accuracy, while the landmark-based MLP delivered a much stronger efficiency profile in latency, FPS, and model size. This makes the image route a stronger full-sample recognition baseline, and the landmark route a more attractive option for constrained, low-cost, edge-oriented settings.

The results also show that robustness must be interpreted carefully. Under strong corruption, the two routes fail in different ways, and the landmark pipeline must be judged together with hand-detector reliability, not only by classifier accuracy after successful detection. Overall, this study supports the feasibility of low-cost touchless document interaction in a controlled static-gesture setting, while also showing that broader validation, end-to-end analysis, and more realistic interface testing are still needed before stronger real-world claims can be made.

Data and Code Availability

The hand-gesture benchmark used in this work is publicly available through Kaggle. The dataset can be accessed at https//www.kaggle.com/datasets/nizamuddinmaitlo/hgr-dataset and the implementation used for duplicate auditing, clean-split evaluation, training, and benchmarking is available at https//www.kaggle.com/code/basitaliharejo/frugal-4-command-touchless-interface/edit.

REFERENCES

[1] S. S. Rautaray and A. Agrawal, "Vision based hand gesture recognition for human computer interaction A survey," *Artificial Intelligence Review*, vol. 43, no. 1, pp. 1–54, 2015, doi 10.1007/s10462-012-9356-9.

[2] N. Mohamed, M. B. Mustafa, and N. Jomhari, "A review of the hand gesture recognition system Current progress and future directions," *IEEE Access*, vol. 9, pp. 157422–157436, 2021, doi 10.1109/ACCESS.2021.3129650.

[3] R. Tripathi and B. Verma, "Survey on vision-based dynamic hand gesture recognition," *The Visual Computer*, vol. 40, no. 9, pp. 6171–6199, 2024, doi 10.1007/s00371-023-03160-x.

[4] C. Cui, M. S. Sunar, and G. Eg Su, "Deep vision-based real-time hand gesture recognition A review," *PeerJ Computer Science*, vol. 11, Art. no. e2921, 2025, doi 10.7717/peerj-cs.2921.

[5] G. C. S. Ruppert, L. O. Reis, P. H. J. Amorim, T. F. de Moraes, and J. V. L. da Silva, "Touchless gesture user interface for interactive image visualization in urological surgery," *World Journal of Urology*, vol. 30, no. 5, pp. 687–691, 2012, doi 10.1007/s00345-012-0879-0.

[6] R. Wipfli, V. Dubois-Ferrière, S. Budry, P. Hoffmeyer, and C. Lovis, "Gesture-controlled image management for operating room A randomized crossover study to compare interaction using gestures, mouse, and third person relaying," *PLOS ONE*, vol. 11, no. 4, Art. no. e0153596, 2016, doi 10.1371/journal.pone.0153596.

[7] N. Madapana, D. Chanci Arrubla, G. T. Gonzalez, L. Zhang, and J. P. Wachs, "Touchless interfaces in the operating room A study in gesture preferences," *International Journal of Human–Computer Interaction*, vol. 39, no. 3, pp. 438–448, 2023, doi 10.1080/10447318.2022.2041896.

[8] W. M. Glinkowski, T. Míścior, and R. Sitnik, "Remote, touchless interaction with medical images and telementoring in the operating room using a Kinect-based application—A usability study," *Applied Sciences*, vol. 13, no. 21, Art. no. 11982, 2023, doi 10.3390/app132111982.

[9] T. Kamijo, A. J. J. M. van Breemen, X. Ma, S. Shanmugam, T. Bel, G. de Haas, *et al.*, "A touchless user interface based on a near-infrared-sensitive transparent optical imager," *Nature Electronics*, vol. 6, no. 6, pp. 451–461, 2023, doi 10.1038/s41928-023-00970-8.

[10] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, *et al.*, "Searching for MobileNetV3," in *Proc. IEEE/CVF Int. Conf. Computer Vision (ICCV)*, 2019, pp. 1314–1324, doi 10.1109/ICCV.2019.00140.

[11] M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, G. Appleton, M. Axton, A. Baak, *et al.*, "The FAIR Guiding Principles for scientific data management and stewardship," *Scientific Data*, vol. 3, Art. no. 160018, 2016, doi 10.1038/sdata.2016.18.

[12] A. Torralba and A. A. Efros, "Unbiased look at dataset bias," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 1521–1528, doi 10.1109/CVPR.2011.5995347.

[13] F. Laakom, J. Raitoharju, J. Nikkanen, A. Iosifidis, and M. Gabbouj, "INTEL-TAU A color constancy dataset," *IEEE Access*, vol. 9, pp. 39560–39567, 2021, doi 10.1109/ACCESS.2021.3064382.

[14] G. Hemrit, G. D. Finlayson, A. Gijsenij, P. Gehler, S. Bianco, B. Funt, *et al.*, "Rehabilitating the ColorChecker dataset for illuminant estimation," in *Proc. IS&T 26th Color and Imaging Conf.*, 2018, pp. 350–353, doi 10.2352/ISSN.2169-2629.2018.26.350.

[15] D. Ulucan, O. Ulucan, and M. Ebner, "CC-NORD A camera-invariant global color constancy dataset," in *Proc. 31st European Signal Processing Conf. (EUSIPCO)*, 2023, pp. 541–545, doi 10.23919/EUSIPCO58844.2023.10289937.

[16] B. Raza, S. Bibi, S. Bibi, and A. Nawaz, "SADA COLOR DATASET (SCD) 9 paper colors × 4 illumination conditions for robust color vision evaluation," *Spectrum of Engineering Sciences*, vol. 4, no. 2, pp. 871–887, 2026, doi 10.5281/zenodo.18844499.

[17] S. Bibi, F. A. Rajput, M. Younis, S. Bibi, and B. Raza, "Vector+SQL retrieval with selectivity workloads Measuring tail latency and quality under filtered Top-K," *VFAST Transactions on Software Engineering*, vol. 14, no. 1, pp. 335–349, 2026, doi 10.21015/vtse.v14i1.2353.