OPTIRSDE: A NOVEL APPROACH WITH TEMPORAL SMOOTHING AND OPTIMIZED FEATURE MATCHING FOR FAST AND ROBUST DEPTH ESTIMATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Depth estimation accuracy over long ranges is a core problem in robotics, maritime autonomy, terrestrial autonomy, and environmental monitoring, where accurate scene understanding is crucial for safe and informed decision-making. Existing monocular solutions suffer a sharp accuracy drop beyond mid-range, with errors of 10–25% at 50–100 m. Recent deep learning–based stereo networks (e.g., FoundationStereo, DSMNet, MonSter, RAFT-Stereo, CREStereo, Selective-Stereo) achieve impressive results on benchmarks but struggle in real-world extended-range scenarios—frequently collapsing at 20–30 m and beyond, where predictions deviate by factors of 2–3× and object-level depth is often lost. In contrast, a calibrated high-quality stereo system can deliver accurate long-range estimates but at the expense of high computational overhead.

We introduce OptiRSDE (Optimized Robust Stereo Depth Estimation), a lightweight yet robust classical computer vision pipeline that integrates disparity refinement, temporal smoothing, and QR-code–based synchronization. OptiRSDE achieves $<$3% error at 50 m and 5–10% at 100 m, substantially outperforming both monocular methods and modern deep learning stereo baselines in real-world conditions. Operating at 5 FPS, while requiring only standard chessboard calibration and YOLO-based object detection for deployment. Temporal smoothing and outlier rejection mitigate depth jitter, producing stable long-range depth at object level. Validated on DrivingStereo Yang et al. (2019) and a custom 1080p stereo dataset, our system demonstrates scalable, real-time, extended-range stereo depth estimation—delivering strong generalization where both monocular and state-of-the-art deep learning methods fail.

1 INTRODUCTION

Real-time, accurate depth estimation is vital for applications like autonomous navigation, robotics, and augmented reality. However, existing vision-based methods exhibit a trade-off between performance and accuracy. Conventional stereo vision approaches, though theoretically precise, are often too slow for real-time use, operating at just 0.2–0.5 fps. To boost frame rates, they often sacrifice accuracy, while modern monocular depth estimation models based on deep learning are too computationally heavy for embedded or mobile systems.

Beyond computational cost, maintaining accuracy and stability in dynamic environments remains a major challenge. Depth precision degrades at long ranges, with typical error rates of 5–10% at 50 meters. Environmental factors worsen this; even minor temperature changes can cause drift and error accumulation of up to 25m at 100m range, requiring frequent recalibration. Temporal instability is another issue, with unsmoothed pipelines showing depth inconsistencies up to ±15% frame-to-frame. This is worsened by stereo camera desynchronization, where a misalignment of 10–12 frames can introduce an additional 10% error at 10–50m distances.

While sensors like LiDAR offer high-accuracy depth data Wang et al. (2020), they come with limitations. High-performance units are too expensive for broad use, while affordable ones have limited range, often below 60 meters. LiDAR also suffers from weather sensitivity and surface reflectiv-

ity issues. Critically, an end-to-end stereo video solution that delivers object-specific depth with dynamic detection and tracking remains largely missing.

To address these challenges, we present a novel, efficient pipeline (Figure 1) that balances accuracy, temporal stability, and real-time performance. Our method synchronizes stereo video streams using QR codes (Figure 5), followed by camera calibration, undistortion, and rectification (Figure 3). We use YOLOv11 Khanam & Hussain (2024) for object detection and BRISK-based feature matching Leutenegger et al. (2011) for robust disparity estimation. Multiple optimizations are introduced to enhance performance. Finally, depth is computed via triangulation (Figure 2), forming a practical, end-to-end solution for real-world deployment.

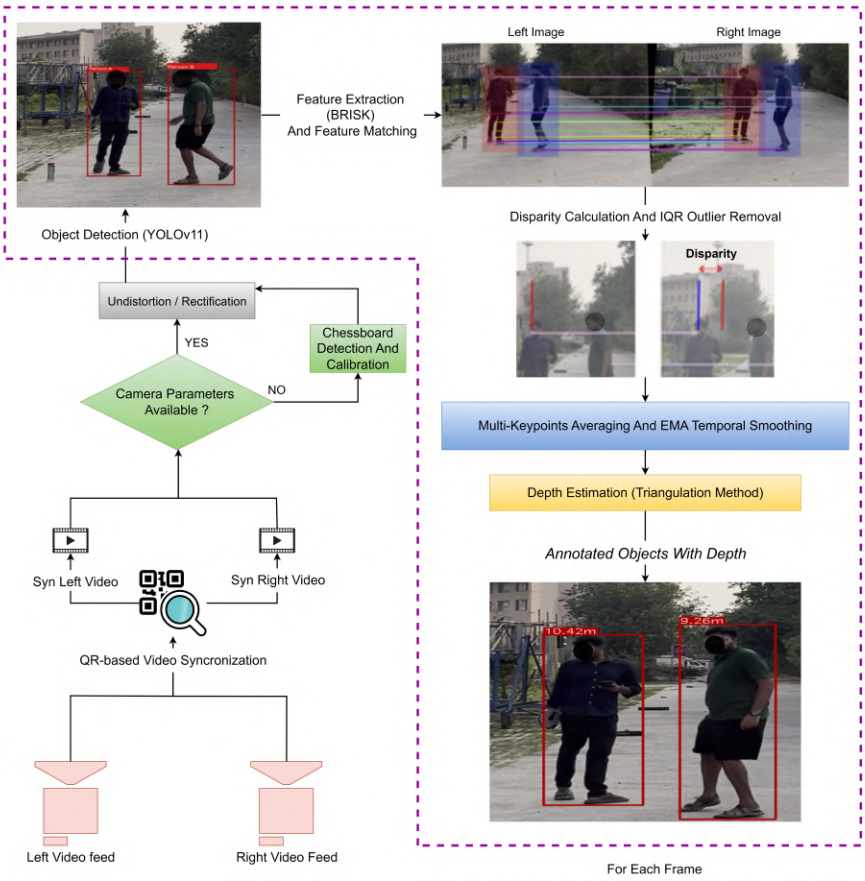

Figure 1: OptiRSDE stereo depth pipeline. Synchronized stereo videos undergo calibration, undistortion, and rectification before object detection and BRISK-based feature matching. Disparity is computed with outlier removal, followed by keypoint averaging and smoothing for depth estimation via triangulation.

2 Related Work

Recent advances in depth estimation target computational efficiency, long-range accuracy, and temporal stability; yet, challenges persist for robust, object-specific, real-time depth in constrained environments.

2.1 Computational Overload and Real-Time Processing

To address the high computational demands in stereo depth rooted in classical correspondence frameworks Scharstein et al. (2001), methods like BiDAStereo Jing et al. (2024) and StereoDiffusion Xu et al. (2024) offer solutions. These rely on robust feature extraction and matching techniques; for

instance, SIFT Lowe (1999) is foundational for establishing correspondences. BiDAStereo employs bidirectional alignment and lightweight recurrent modules, while StereoDiffusion integrates optical-flow-warped disparities into a diffusion model for high-speed, consistent predictions. However, their utility in low-power or embedded systems remains limited and needs further optimization.

2.2 Long-Range Depth Accuracy

Long-range depth estimation progresses with trinocular and attention models. For example, Yoshihara et al. (2025) localizes vessels up to 2.5 km; GatedStereo Walz et al. (2023) fuses active gated and stereo cues for enhanced performance. While accurate, these systems remain expensive, difficult to deploy, or unsuitable for real-time use. Monocular models like Depth Anything Yang et al. (2024) progress, yet stereo methods like STTR Li et al. (2021a) remain superior for long-range tasks. Large datasets like DrivingStereo Yang et al. (2019) and KITTI Geiger et al. (2012) advance robust mid-to-long range model training. Despite potential, dynamic environment robustness remains challenging.

2.3 Calibration and Synchronization

Accurate stereo depth estimation depends on effective calibration and synchronization. Solutions like BiDAStereo and Li et al. (2021a) use cross-frame alignment, disparity refinement to mitigate misalignment. Optical flow models like CODD Li et al. (2021b) compensate for dynamic motion and drift. Frequent recalibration under changing conditions remains unresolved practically.

2.4 Temporal Stability

Temporal inconsistencies degrade real-time depth quality. Robust outlier rejection, often via RANSAC Fischler & Bolles (1987), mitigates inconsistencies and noise. Building on linear filtering theories Basar (2001), techniques like dual-space disparity refinement Zeng et al. (2025) and DynamicStereo Karaev et al. (2023) significantly reduce variance. StereoDiffusion Xu et al. (2024) incorporates temporal priors for smoothness. While these models reduce jitter, stability in dynamic or low-texture scenes needs improvement.

2.5 Object-Level Depth Estimation

Object-specific depth estimation gains traction beyond global maps. Systems by Yoshihara et al. (2025) and Zheng et al. (2022) integrate object detection with disparity for tailored depth, e.g., maritime vessels. Transformer pipelines like DynamicStereo Karaev et al. (2023) support object-level consistency by fusing spatial-temporal features. These works emphasize coupling semantic understanding with depth in complex, cluttered scenes.

3 Methodology

The proposed depth estimation pipeline is designed to overcome critical challenges in stereo vision systems, including calibration inaccuracies, unreliable feature detection, instability across frames, and high computational demands. Our approach builds upon and refines the foundational principles of stereo depth estimation.

Classic Depth Estimation Pipeline Overview:

- **Calibration**: It establishes intrinsic and extrinsic camera parameters. Intrinsic parameters correct lens distortions (radial, tangential). Extrinsic parameters determine relative camera pose (rotation, translation).
- **Rectification**: Stereo image pairs are transformed for horizontal epipolar line alignment. This rectification Fusiello et al. (2000) simplifies correspondence search as matching pixels lie on the same row.
- **Correspondence Matching**: Feature points are identified and matched across stereo pairs. These matched points generate a disparity map Fua (1991), encoding relative displacement.

3.1 Stereo Video Synchronisation

Temporal alignment of left and right video streams is critical for accurate stereo depth estimation. As our system lacked hardware synchronization capabilities, we implemented a robust software-based method to ensure precise frame-to-frame correspondence. This approach leverages a series of unique visual markers embedded within the video feed to determine and correct any temporal offset. A full description of this synchronization technique is provided in Appendix A.

3.2 Stereo Camera Calibration

Accurate 3D reconstruction and depth estimation depend on precise stereo camera calibration to determine intrinsic and extrinsic parameters. This process is performed once, after the physical setup is fixed, and must be repeated only if the camera positions change.

Stereo calibration is performed using a chessboard pattern held in front of both cameras. The calibrated parameters are then used for all future video inputs.

This process is shown in detail in Figure 3.

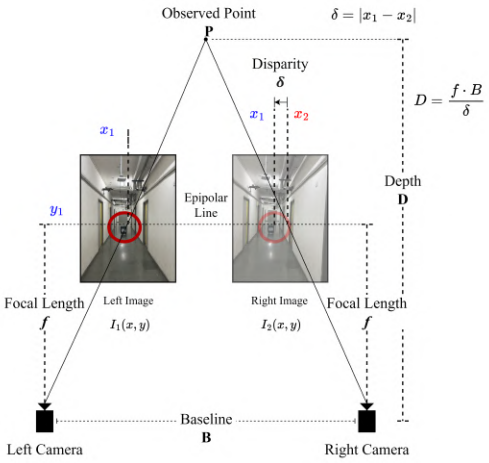

Figure 2: Schematic representation of stereo vision for depth estimation. An observed 3D point projects onto corresponding points in the left and right images, with a horizontal shift (disparity) $\delta = |x_1 - x_2|$. Depth D is computed using the known focal length f, baseline B, and disparity: $D = \frac{f \cdot B}{\delta}$.

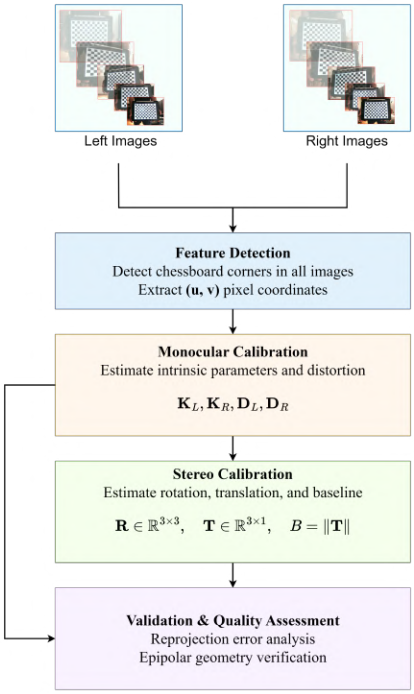

Figure 3: Overview of the stereo camera calibration pipeline. Chessboard corners are detected in synchronized left and right images to extract pixel coordinates. Monocular and stereo calibration estimate intrinsic and extrinsic parameters, followed by validation through reprojection error and epipolar geometry checks.

3.3 Depth Estimation

Our system estimates depth using disparity-to-depth triangulation (Figure 2) for each detected object. BRISK Leutenegger et al. (2011) is employed for keypoint detection, offering enhanced resilience in low-texture environments compared to traditional methods like ORB Rublee et al. (2011).

This keypoint-based approach is crucial for achieving high accuracy in real-time stereo depth estimation. By restricting keypoint detection to object-specific Regions of Interest (ROIs), the system efficiently prioritizes relevant features for disparity computation. This optimization not only significantly reduces computational load by avoiding unnecessary processing across the entire image but also substantially improves the precision of correspondence matching by adaptively expanding the search space for stereo parallax solely within relevant bounding boxes, thereby enhancing overall system robustness and speed.

Erroneous matches are filtered using the Inter-Quartile Range (IQR). Valid disparities from multiple matched keypoints within each object's bounding box are then averaged for robust depth estimation, similar to harmonic depth averaging Yoshihara et al. (2025), but with a slight numerical precision advantage.

To mitigate frame-to-frame depth fluctuations, Exponential Moving Average (EMA) temporal smoothing is applied every 10 frames. This stabilizes predictions and suppresses noise from spurious and transient errors.

3.4 Efficiency-Oriented Pipeline Optimizations

Several optimizations are integrated into the pipeline to balance computational efficiency and estimation accuracy:

- **Keypoint detection is localized** to object bounding boxes through masking, dramatically reducing processing time by avoiding exhaustive search.
- **Feature matching is further constrained** to filtered keypoints of corresponding image regions, yielding $16\times$ speed improvement for this particular component.
- **The FSRCNN-based super-resolution module** Dong et al. (2016), previously used to enhance image detail in Zheng et al. (2022), was eliminated. The use of BRISK compensates for this removal, maintaining robust performance while reducing computational overhead.

3.5 Temporal Stability in Dynamic Environments

To ensure temporal consistency in depth estimation within dynamic scenes, the system integrates two complementary techniques:

- Multi-keypoint disparity averaging within object regions, reducing sensitivity to individual erroneous matches.
- EMA-based temporal smoothing, attenuating sudden depth variations from detection noise or transient mismatches. The Exponential Moving Average (EMA) Yu et al. (2020) $\bar{D}_t$ is calculated using the current disparity P_t and the previous smoothed disparity $\bar{D}_{t-1}$:

$$\bar{D}_t = \alpha \cdot P_t + (1 - \alpha) \cdot \bar{D}_{t-1}$$

  This strategy, detailed in Algorithm 1, significantly improves the stability and reliability of depth predictions, even under challenging environmental conditions.

4 Evaluation

To evaluate our stereo depth estimation system, we benchmarked OptiRSDE against monocular **(MiDaS)** Ranftl et al. (2022) and stereo (our implementation of Zheng's method Zheng et al. (2022), **BSV Ship**) Zheng et al. (2022) baselines for depth accuracy. Performance comparison is done only between our method and Zheng's method, since it is the only other object-level binocular stereo depth estimation pipeline to the best of our knowledge. We evaluated OptiRSDE on our self-collected dataset (ground truth from distance markers) and DrivingStereo's test images Yang et al. (2024), since it provides LiDAR-based depth maps as ground truth. For our dataset, a reference grid with known ground-truth distances (at 10-meter intervals, using a precise device) was used. Tests were conducted on subjects across **5–100m distances**.

Algorithm 1 Temporal Smoothing for Disparity Estimation

Input: Left and right frames $F_L, F_R \in \mathbb{R}^{H \times W \times 3}$, current frame number $t \in \mathbb{N}$, smoothing factor $\alpha \in [0, 1]$, expiration threshold $T_{\text{expire}} \in \mathbb{N}$, persistent object state map S.
Output: Detected objects B (list of bounding boxes), updated disparities D_{raw}.

$B \leftarrow \text{TrackObjects}(F_L)$
$D_{\text{raw}} \leftarrow \text{CalculateDisparities}(F_L, F_R, B)$
$M_{\text{ID}\to\text{idx}} \leftarrow \emptyset$
for each box $b_i \in B$ at index i **do**
  $id_{\text{obj}} \leftarrow \text{GetID}(b_i)$
  $M_{\text{ID}\to\text{idx}}[id_{\text{obj}}] \leftarrow i$
  if id_{obj} not in S **then**
    $S[id_{\text{obj}}] \leftarrow \{\text{disparity: null}, \text{frame: } t\}$
  end if
end for
for each id_{obj} in keys of S **do**
  $s \leftarrow S[id_{\text{obj}}]$
  if $s.\text{frame} + T_{\text{expire}} \leq t$ **then**
    Remove id_{obj} from S
    continue
  end if
  if id_{obj} in $M_{\text{ID}\to\text{idx}}$ **then**
    $i \leftarrow M_{\text{ID}\to\text{idx}}[id_{\text{obj}}]$
    $d_{\text{current}} \leftarrow D_{\text{raw}}[i]$
    if $s.\text{disparity} = \text{null}$ **then**
      $s.\text{disparity} \leftarrow d_{\text{current}}$
    else if $d_{\text{current}} = \text{null}$ **then**
      $D_{\text{raw}}[i] \leftarrow s.\text{disparity}$
    else
      $d_{\text{smooth}} \leftarrow \alpha \cdot d_{\text{current}} + (1 - \alpha) \cdot s.\text{disparity}$
      $s.\text{disparity} \leftarrow d_{\text{smooth}}$
      $D_{\text{raw}}[i] \leftarrow d_{\text{smooth}}$
    end if
    $s.\text{frame} \leftarrow t$
  end if
end for
return B, D_{raw}

4.1 Experimental Settings

4.2 Hardware:

The system runs on an AMD Ryzen 5 4600H CPU with 8GB RAM and an NVIDIA GeForce GTX 1650 Ti GPU (4GB VRAM). The stereo vision setup consists of dual 1080p cameras mounted with a fixed 35 cm baseline.

4.3 Software:

The system uses Python 3.12 on Linux Mint 22.1 and OpenCV 4.5 for image processing. PyTorch 2.5.1 integrates YOLOv11 for object detection, leveraging CUDA 12.6.

4.4 Camera Baseline

Our system uses a 35 cm baseline, balancing flexibility and accuracy, with potential for further extension.

4.5 CAMERA CALIBRATION

We used a **9×6 chessboard** across **2 to 20 meters**. Rectification error between stereo pairs was reduced to under 0.75 pixels, ensuring reliable disparity estimation.

4.6 OBJECT DETECTION AND STEREO MATCHING

Up to 512 feature points were extracted per frame; only those with ≤ 50 Hamming distance Liu et al. (2018) were retained.

4.7 TEMPORAL SMOOTHING

To ensure consistency, an **Exponential Moving Average (EMA)** ($\alpha = 0.3$) was applied every 10 frames. Outlier disparities were removed using **Inter-Quartile Range (IQR)** Takiar (2023) filtering, reducing noise and stabilizing predictions.

5 RESULTS

We compared our method **(OptiRSDE)** against monocular **(MiDaS)** Ranftl et al. (2022) and stereo **(BSV Ship)** Zheng et al. (2022) baselines across two metrics: depth accuracy and speed. Note that we used our own implementation of Zheng's BSV Ship Depth Estimation methodology, since no code is available for it publicly. Tests were conducted on 1080p stereo video at **5–100m distances**. All the results are generated from a single run of the algorithm over each video.

Method	Err@50m	Err@100m
Monocular (MiDaS)	12% (4%)	25% (9%)
Stereo (BSV Ship)	4% (1.4%)	10% (6.1%)
OptiRSDE	**2.8% (0.5%)**	**7.5% (1.9%)**

Table 1: Depth estimation accuracy comparison. Mean error values with standard deviation in parentheses.

Method	FPS
BSV Ship	0.3 (0.02)
OptiRSDE	**5.38 (0.08)**

Table 2: Execution performance comparison. Mean FPS values with standard deviation in parentheses.

5.1 ANALYSIS

We evaluated OptiRSDE's depth estimation accuracy across multiple distances. Estimated distances were validated against calibrated ground truth using our internal dataset and DrivingStereo test images, where LiDAR-based depth maps served as ground truth. Visual results (Figure 4) and quantitative data (Tables 3, 4) are presented. OptiRSDE was compared to our implementation of Zheng's method (*BSV Ship*), MonSter Cheng & et al. (2025), DSMNet Zhang et al. (2020), FoundationStereo Wen et al. (2025), RAFT-Stereo Lipson et al. (2021), CREStereo Li et al. (2022), and Selective-IGEV Wang et al. (2024). Results consistently demonstrate OptiRSDE's superior performance, achieving significantly lower estimation errors, especially at larger distances.

5.2 KEY FINDINGS

- **Accuracy:** OptiRSDE reduces errors by 77% vs. MiDaS and 30% vs. BSV Ship at 50m and by 70% vs. MiDaS and 25% vs. BSV Ship at 100m. (Table 1)
- **Speed:** Achieves > 5 FPS—10-20× faster than traditional stereo methods. (Table 2)
- **Robustness:** Consistent keypoint detection ensures continuous depth estimation, overcoming frequent detection failures observed in BSV Ship. (See NKD in Table 3)
- **Stability:** Temporal smoothing cuts depth jitter compared to unsmoothed baselines. (Refer to supplementary videos)

Method	10m	20m	30m	40m	50m	60m
MonSter	-0.68	-0.6	+14.22	+25.4	-2.2	+10.76
DSMNet	-0.8	-1.68	+7.21	+17.65	+15.17	+30.69
FoundationStereo	-0.77	-0.81	**+0.04**	+4.79	+16.81	+29.89
RAFT-Stereo	-0.75	-0.87	-1	+1.19	+10.38	+10.29
CREStereo	-5.17	-6.88	-1.24	+14.92	+16.7	+82.35
Selective-IGEV	**-0.53**	**+0.03**	-0.77	+5.45	+1.78	+6.08
BSV Ship	+0.68	+74.65	-2.9	-4.36	NKD	+5.36
OptiRSDE (Ours)	-0.62	-0.59	0.72	**-0.04**	**+1.53**	**-1.77**

Table 3: Depth estimation errors (in meters) for different methods. The values in the column headers represent ground truth distances (in meters). All listed methods are different approaches for estimating depth from calibrated left–right image pairs. **Bold** represents the smallest errors and underline represents the second smallest errors at a particular distance. **NKD** indicates cases where no keypoints were detected.

Method	8.20m	16.70m	24.19m	44.26m	47.16m
MonSter	+2.72	+41.24	+0.17	+1.91	**-0.17**
DSMNet	+0.54	-0.45	+0.37	+1.03	+0.47
FoundationStereo	+0.46	0.5	+0.44	-0.75	0.61
RAFT-Stereo	+0.81	-0.26	+0.51	-0.29	+1.07
CREStereo	-5.62	-10.93	-15.32	-21.57	-22.32
Selective-IGEV	+3.02	+0.44	+5.04	+3.05	+3.29
BSV Ship	-0.16	NKD	+1.23	+47.26	-1.4
OptiRSDE (Ours)	**-0.01**	**-0.02**	**+0.08**	**+0.14**	+0.30

Table 4: Depth estimation errors on DrivingStereo's test images. Same structure as Table 3

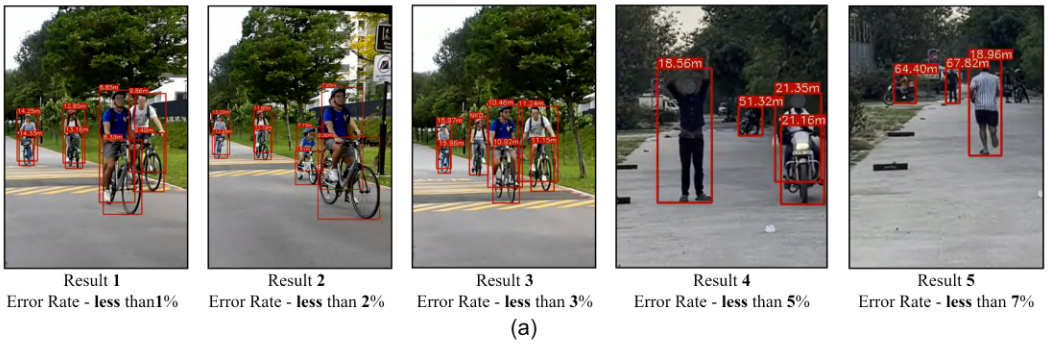

Figure 4: Results of depth estimation up to 120m (a) from our own dataset. (b) from DrivingStereo's test images.

6 ABLATION STUDY

To evaluate the impact of individual components in our proposed pipeline, we conducted an ablation study by systematically disabling key modules and measuring their effect on depth estimation accuracy and computational efficiency.

- **Temporal Smoothing:** Removing the Exponential Moving Average (EMA) filter increased depth jitter, particularly at distances beyond 30m. Frame-to-frame variance rose from ±2% to ±10%, confirming EMA's critical role in stabilizing long-range estimates.
- **Feature Detection and Matching:** Replacing BRISK with ORB significantly decreased the robustness of keypoint detection. The results started showing more *No keypoint detected* errors, as ORB has difficulty detecting keypoints in relatively low-texture regions.
- **Synchronization:** Without QR-based synchronization, depth errors spiked up to 10% at 20m due to misaligned stereo pairs.
- **Computational Optimizations:** Disabling masking before keypoint detection, removing our keypoint matching optimization, or adding back FSRCNN significantly reduced the FPS (Table 5), while removing IQR-based outlier rejection increased depth variance by 30%.

Method	**FPS**
Without KP detection masking	4.85 (0.11)
Without KP matching optimization	4.49 (0.08)
With FSRCNN	1.53 (0.03)
No ablation	**5.38 (0.08)**

Table 5: Execution performance after ablation. Mean FPS values with standard deviation in parentheses.

7 CONCLUSION

We presented OptiRSDE, a robust stereo depth estimation pipeline integrating temporal smoothing, optimized feature matching, and efficient synchronization to achieve high long-range accuracy and performance. Key contributions include its long-range precision, where pixel disparity refinement and harmonic averaging enable less than 10% depth error up to 100 meters; high performance, with several optimizations delivering about 5 FPS; and minimal setup, leveraging QR-based synchronization and chessboard calibration for simplified deployment. Ablation studies validated the necessity of each component, while benchmarks consistently demonstrated OptiRSDE's superiority over monocular and previous stereo baselines in terms of accuracy, speed, and robustness. The system's precise, long-range, fast, and temporally stable depth perception offers significant utility, poised to enhance autonomous driving (for accident detection and collision avoidance) and industrial safety systems. It can also support diverse industrial applications, including advanced robotics, object-aware automation, and sophisticated surveillance requiring accurate 3D scene understanding. Future work may involve exploring edge-device deployment and multi-sensor fusion (e.g., LiDAR) for robust performance in adverse weather conditions.

REFERENCES

Tamer Basar. *A New Approach to Linear Filtering and Prediction Problems*. 2001. doi: 10.1109/9780470544334.ch9.

J. Cheng and et al. MonSter: Marry Monodepth to Stereo Unleashes Power. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6273–6282, 2025. doi: 10.1109/CVPR52734.2025.00588.

Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling (eds.), *Computer*

Vision – ECCV 2016, pp. 391–407, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46475-6.

Martin A. Fischler and Robert C. Bolles. *Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1987.

Pascal Fua. A Parallel Stereo Algorithm that Produces Dense Depth Maps and Preserves Image Features. January 1991.

Andrea Fusiello, Emanuele Trucco, and Alessandro Verri. A compact algorithm for rectification of stereo pairs. *Mach. Vision Appl.*, 12(1):16–22, July 2000. ISSN 0932-8092.

Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are We Ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3354–3361, 2012. doi: 10.1109/CVPR.2012.6248074.

Junpeng Jing, Ye Mao, and Krystian Mikolajczyk. Match-stereo-videos: Bidirectional alignment for consistent dynamic stereo matching. 2024.

Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Dynamicstereo: Consistent dynamic depth from stereo videos. *CVPR*, 2023.

Rahima Khanam and Muhammad Hussain. Yolov11: An overview of the key architectural enhancements, 2024. URL https://arxiv.org/abs/2410.17725.

Stefan Leutenegger, Margarita Chli, and Roland Y. Siegwart. BRISK: Binary Robust Invariant Scalable Keypoints. In *2011 International Conference on Computer Vision (ICCV)*, pp. 2548–2555, 2011. doi: 10.1109/ICCV.2011.6126542.

Jiankun Li, Peisen Wang, Pengfei Xiong, Tao Cai, Ziwei Yan, Lei Yang, Jiangyu Liu, Haoqiang Fan, and Shuaicheng Liu. Practical Stereo Matching via Cascaded Recurrent Network with Adaptive Correlation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16242–16251, 2022. doi: 10.1109/CVPR52688.2022.01578.

Zhaoshuo Li, Xingtong Liu, Nathan Drenkow, Andy Ding, Francis X. Creighton, Russell H. Taylor, and Mathias Unberath. Revisiting stereo depth estimation from a sequence-to-sequence perspective with transformers, 2021a. URL https://arxiv.org/abs/2011.02910.

Zhaoshuo Li, Wei Ye, Dilin Wang, Francis X. Creighton, Russell H. Taylor, Ganesh Venkatesh, and Mathias Unberath. Temporally consistent online depth estimation in dynamic scenes. *CoRR*, abs/2111.09337, 2021b. URL https://arxiv.org/abs/2111.09337.

Lahav Lipson, Zachary Teed, and Jia Deng. RAFT-Stereo: Multilevel Recurrent Field Transforms for Stereo Matching. In *International Conference on 3D Vision (3DV)*, 2021.

Yanli Liu, Heng Zhang, Hanlei Guo, and Neal N. Xiong. A FAST-BRISK Feature Detector with Depth Information. *Sensors*, 18(11):3908, November 2018. doi: 10.3390/s18113908. URL https://www.mdpi.com/1424-8220/18/11/3908.

D.G. Lowe. Object Recognition from Local Scale-Invariant Features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, pp. 1150–1157, 1999. doi: 10.1109/ICCV.1999.790410.

René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(03):1623–1637, March 2022. doi: 10.1109/TPAMI.2020.3019967. URL https://doi.ieeecomputersociety.org/10.1109/TPAMI.2020.3019967.

Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. ORB: An Efficient Alternative to SIFT or SURF. In *2011 International Conference on Computer Vision (ICCV)*, pp. 2564–2571, 2011. doi: 10.1109/ICCV.2011.6126544.

D. Scharstein, R. Szeliski, and R. Zabih. A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms. In *Proceedings IEEE Workshop on Stereo and Multi-Baseline Vision (SMBV)*, pp. 131–140, 2001. doi: 10.1109/SMBV.2001.988771.

Ramnath Takiar. A new method to identify the outliers based on the inter quartile range. 11:103–114, 10 2023. doi: 10.33329/bomsr.11.4.103.

Stefanie Walz, Mario Bijelic, Andrea Ramazzina, Amanpreet Walia, Fahim Mannan, and Felix Heide. Gated Stereo: Joint Depth Estimation from Gated and Wide-Baseline Active Stereo Cues. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13252–13262. IEEE, June 2023. doi: 10.1109/CVPR52729.2023.01273.

Xianqi Wang, Gangwei xu, Hao Jia, and Xin Yang. Selective-Stereo: Adaptive Frequency Information Selection for Stereo Matching. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19701–19710, 2024. doi: 10.1109/CVPR52733.2024.01863.

Yan Wang, Wei-Lun Chao, Divyansh Garg, Bharath Hariharan, Mark Campbell, and Kilian Q. Weinberger. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving, 2020. URL https://arxiv.org/abs/1812.07179.

Bowen Wen, Matthew Trepte, Joseph Aribido, Jan Kautz, Orazio Gallo, and Stan Birchfield. FoundationStereo: Zero-Shot Stereo Matching. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5249–5260, 2025. doi: 10.1109/CVPR52734.2025.00495.

Haozheng Xu, Chi Xu, and Stamatia Giannarou. StereoDiffusion: Temporally Consistent Stereo Depth Estimation with Diffusion Models . In *Proceedings of Medical Image Computing and Computer Assisted Intervention – MICCAI 2024*, volume LNCS 15006. Springer Nature Switzerland, October 2024.

Guorun Yang, Xiao Song, Chaoqin Huang, Zhidong Deng, Jianping Shi, and Bolei Zhou. DrivingStereo: A Large-Scale Dataset for Stereo Matching in Autonomous Driving Scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 899–908, 2019. doi: 10.1109/CVPR.2019.00099.

Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth Anything V2. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2024. URL https://openreview.net/forum?id=cFTi3gLJ1X.

Kotaro Yoshihara, Shigehiro Yamamoto, and Takeshi Hashimoto. On-ship trinocular stereo vision: An experimental study for long-range high-accuracy localization of other vessels. *Journal of Marine Science and Engineering*, 13(1):115, 2025. ISSN 2077-1312. doi: 10.3390/jmse13010115. URL https://www.mdpi.com/2077-1312/13/1/115.

Jaehong Yu, Seoung Bum Kim, Jinli Bai, and Sung Won Han. Comparative study on exponentially weighted moving average approaches for the self-starting forecasting. *Applied Sciences*, 10 (20), 2020. ISSN 2076-3417. doi: 10.3390/app10207351. URL https://www.mdpi.com/2076-3417/10/20/7351.

Jiaxi Zeng, Chengtang Yao, Yuwei Wu, and Yunde Jia. Temporally Consistent Stereo Matching. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024 - 18th European Conference*, volume LNCS 14751, pp. 341–359, Germany, September 2025. Springer Science and Business Media Deutschland GmbH. ISBN 9783031727504. doi: 10.1007/978-3-031-72751-1_20.

Feihu Zhang, Xiaojuan Qi, Ruigang Yang, Victor Prisacariu, Benjamin Wah, and Philip Torr. Domain-invariant Stereo Matching Networks. In *Europe Conference on Computer Vision (ECCV)*, 2020.

Yuanzhou Zheng, Peng Liu, Long Qian, Shiquan Qin, Xinyu Liu, Yong Ma, and Ganjun Cheng. Recognition and depth estimation of ships based on binocular stereo vision. *Journal of Marine Science and Engineering*, 10(8):1153, 2022. ISSN 2077-1312. doi: 10.3390/jmse10081153. URL https://www.mdpi.com/2077-1312/10/8/1153.

8 APPENDIX

This Appendix consists of an in-depth discussion of the following topics:

- Appendix A: QR Synchronization
- Appendix B: Keypoint detection masking optimization.
- Appendix C: Keypoint matching optimization.
- Appendix D: Visual comparison of results.

Due to the supplementary material size limit, images in this document will be of lower quality. The dataset samples and the results are also not provided in the zip file. Link for the full supplementary material is provided in the `README.txt` file.

9 APPENDIX A: QR SYNCHRONIZATION

Accurate stereo video frame synchronisation is crucial for reliable depth estimation. Since we did not have any cameras available with hardware synchronization properties, we devised a novel method to do automated software-based synchronization. A **QR code-based synchronization** method was implemented to precisely align left and right video streams.

At the start of each stereo video, sequential QR codes (e.g., 1, 2, 3, ...) are embedded into frames at the video's frame rate. During preprocessing, a QR code reader extracts these numerical values from each frame of both videos. Comparing these numbers reveals the frame offset (e.g., 14 or 20 frames) between streams. Based on this offset, unmatched or extra frames are discarded, resulting in perfectly aligned video streams. This significantly improves subsequent stereo processing accuracy by eliminating temporal misalignment and ensuring frame-to-frame correspondence, as shown in Figure 5.

Our method is designed specifically for stereo videos and works best when QR-based synchronization is used. For stereo videos without QR codes, the pipeline can still operate; however, synchronization errors typically around 10 to 12 frames can arise, which may significantly increase the depth estimation error rate.

Frame No.	F83	F84	F85	F86	F87	F88	F89	F90
QrReader No.	10	11	12	13	14	15	16	17
Left Frames								
Right Frames								
QrReader No.	10	11	12	13	14	15	16	17
Frame No.	F97	F98	F99	F100	F101	F102	F103	F104

Figure 5: QR-based synchronization of stereo video frames. Sequential QR codes embedded in each frame allow precise alignment of left and right video streams by matching decoded numbers. This eliminates temporal offsets, ensuring accurate frame correspondence for improved depth estimation.

10 APPENDIX B: KEYPOINT DETECTION MASKING OPTIMIZATION

To optimize the computational efficiency of the BRISK keypoint detection algorithm within our OptiRSDE pipeline, we implemented a masking strategy. This targeted approach focuses keypoint

detection on relevant areas of the image, which significantly reduces processing time without compromising the integrity of the feature matching process.

10.1 Overview of Keypoint Detection

Keypoint detection algorithms, such as BRISK, typically operate by exhaustively scanning the entire image to identify distinctive features. These features are robust to various image transformations and are crucial for tasks like feature matching. In a standard, unmasked approach, the algorithm processes every pixel, leading to considerable computational overhead, especially for high-resolution video. This results in a distribution of keypoints across the entire scene, including background elements that may not be relevant for object-level analysis. An example of such unmasked keypoint detection is presented in Figure 6.

Figure 6: Illustration of unmasked keypoint detection and matching in left and right frames (separated by the white border). The left frame shows the detected objects and the estimated depths. Note the distribution of keypoints across both relevant objects and irrelevant background areas.

10.2 Masked Keypoint Detection for Efficiency

Instead of processing the entire image, we leverage the results from an initial object detection step (using YOLOv11 in our main pipeline) to define specific Regions of Interest (ROIs). For the purpose of object-level depth estimation, keypoints are primarily needed on and around the detected objects, not in static, irrelevant background regions of the image.

10.3 Mask Generation

For the left stereo image, a binary mask is generated directly from the bounding boxes provided by the object detector. This mask effectively restricts the BRISK keypoint detection algorithm to these object-containing regions, thereby substantially reducing the search space.

For the corresponding right stereo image, the masking strategy is critically adapted to account for the expected horizontal parallax shift inherent in stereo vision. An object appearing at a certain horizontal position in the left image will appear shifted to the left in the right image. Therefore, in addition to the regions defined by the left image's bounding boxes, we strategically expand the masked region in the right image to include the entire left side of each corresponding bounding box. This expansion is a crucial step to ensure that corresponding keypoints, potentially shifted due to disparity, are still well within the detection zone.

The effect of this masked keypoint detection is visually represented in Figure 7. Note that the blacked-out regions in the figure are for illustrative purposes only, and the mask is passed to the BRISK algorithm itself rather than being applied to the image pixels.

10.4 Performance Impact

By focusing keypoint detection solely on these masked regions, the number of pixels that BRISK must process is drastically reduced. This direct reduction in computational load translates into a significant speedup for the keypoint detection phase of the pipeline. As detailed in the main paper's ablation study, disabling this keypoint detection masking resulted in a noticeable decrease in the overall Frames Per Second (FPS), confirming its indispensable role in achieving the high-speed efficiency of OptiRSDE. This optimization exemplifies how domain-specific knowledge can be effectively leveraged for substantial performance gains in computer vision tasks.

Figure 7: Illustration of masked keypoint detection and matching in left and right frames (separated by the white border). The left frame shows the detected objects and the estimated depths. Keypoints are concentrated within the object bounding boxes in the left frame and their expanded parallax regions in the right frame, demonstrating reduced processing area.

11 Appendix C: Keypoint Matching Optimization

A primary computational bottleneck in keypoint-based stereo vision is the exhaustive matching process. A brute-force approach compares every keypoint in the left image against every keypoint in the right, leading to a combinatorial explosion of pairwise comparisons. This is computationally prohibitive for real-time applications, especially in dynamic scenes where low latency is critical. Our work, OptiRSDE, addresses this challenge by implementing an efficient, object-centric matching strategy that significantly curtails the search space.

Instead of a global, exhaustive search, our method localizes the matching process to relevant regions of interest (ROIs) defined by object bounding boxes. This targeted methodology drastically reduces pairwise comparisons by focusing computational effort only where needed. The cumulative effect of this optimization yielded a remarkable 16-fold performance improvement for keypoint detection and matching. In our experiments with a 10-second, 1080p video, this optimization reduced processing time from 110 seconds to a mere 6.9 seconds. This significant acceleration transforms a demanding task into a tractable one, paving the way for efficient, near real-time depth estimation for dynamic objects.

11.1 Implementation Details

The core of our optimization is a two-stage filtering process. First, keypoints are detected across both the left and right frames using the BRISK algorithm. However, only keypoints within the predefined bounding boxes for each object are retained. All keypoints outside these regions are discarded, immediately reducing the dataset for the more intensive matching algorithm.

A critical aspect of this implementation is the correct handling of stereo parallax, similar to keypoint detection masking. Due to baseline separation, an object's projection in the right image shifts horizontally to the left. To account for this, the ROI in the right image is expanded, but unlike the masking stage, it is done for each object individually. While vertical bounds remain, the horizontal search area spans from the right edge of the box to the left edge of the image frame. This ensures all potential corresponding keypoints are included for matching, regardless of depth and disparity.

Only after this filtering and region adjustment is the brute-force matching algorithm, configured to use Hamming distance, applied. The matching is not performed globally on the filtered keypoints. Instead, it is done on a per-object basis, comparing keypoints from a left ROI exclusively against those in the corresponding adjusted right ROI. This targeted, object-by-object matching is a cornerstone of our optimized pipeline, ensuring both high speed and accuracy.

12 Appendix D: Visual comparison of results

In Figure 6, unmasked keypoint detection and stereo matching are shown. This visualization of stereo pairs shows keypoints scattered across both object regions and irrelevant backgrounds, potentially introducing noise and ambiguity in depth estimation. In contrast, Figure 7 showcases the stark contrast of masked keypoint detection, with keypoints tightly concentrated within object bounding

boxes and their projected parallax regions in the right frame, demonstrating reduced processing areas and more focused feature matching.

Moreover, a set of distance visualization frames is shown as Result 1, Result 2, and Result 3 in Figure 8, Figure 9, and Figure 10 simultaneously, where we compare depth estimation errors between OptiRSDE and the BSV Ship's baseline. The results clearly show that OptiRSDE achieves more accurate depth predictions across varying distances, including at challenging longer ranges where BSV Ship shows NKD (No Keypoint Detection) to detect keypoints. These comparisons visually and quantitatively highlight the spatial efficiency and superior depth accuracy of OptiRSDE.

Additionally, Result 4 in Figure 11 shows depth estimation errors on DrivingStereo's test images, further reinforcing the generalizability and performance of OptiRSDE on unseen data. The results in both datasets consistently demonstrate OptiRSDE's advantages over existing baseline methods, particularly in terms of its robustness and precision across various and challenging distance ranges.

The visualized stereo frame pairs correspond to the Result section of the main paper, including the depth estimation errors and the comparison between OptiRSDE and the BSV baseline methods. These visualizations offer a direct, intuitive understanding of the performance differences between the methods, giving a detailed visual comparison of the results presented in the Results section of the paper.

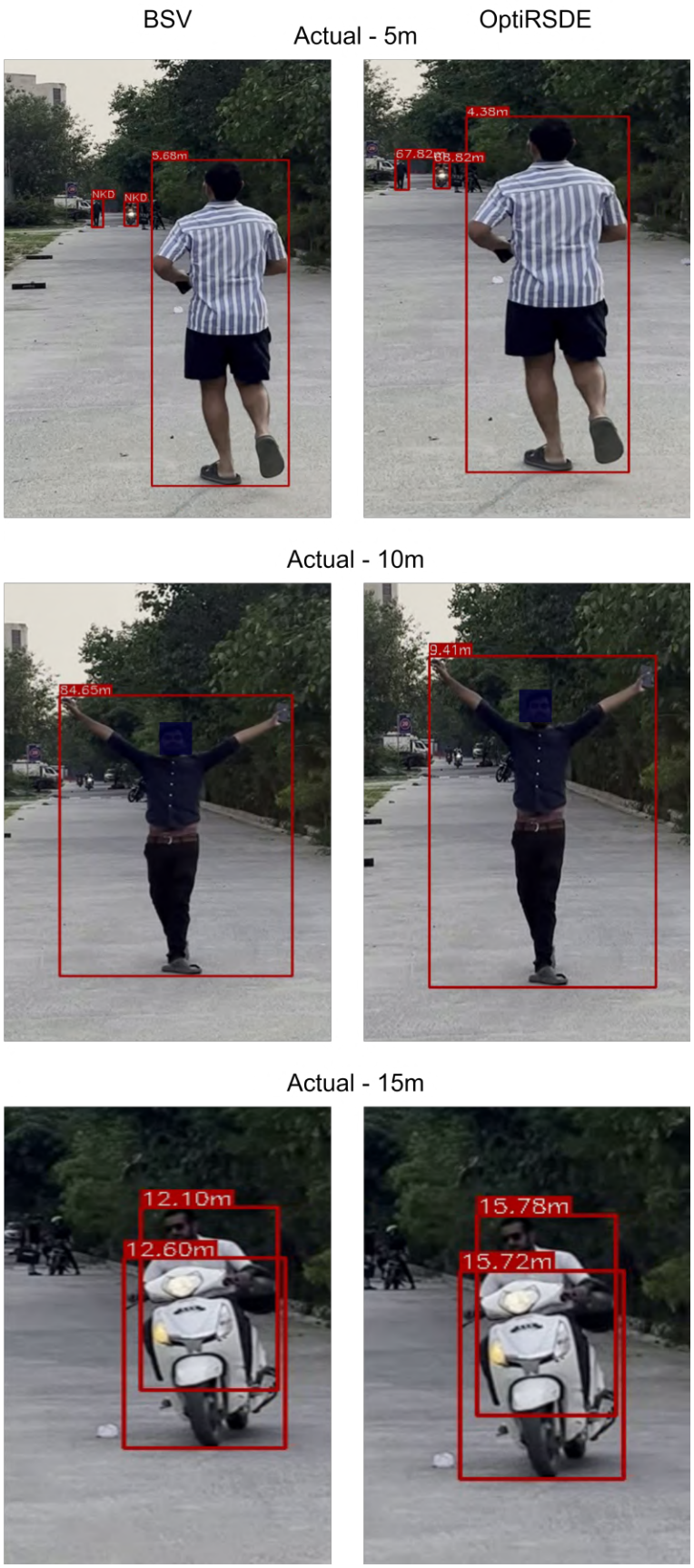

Figure 8: Result 1.

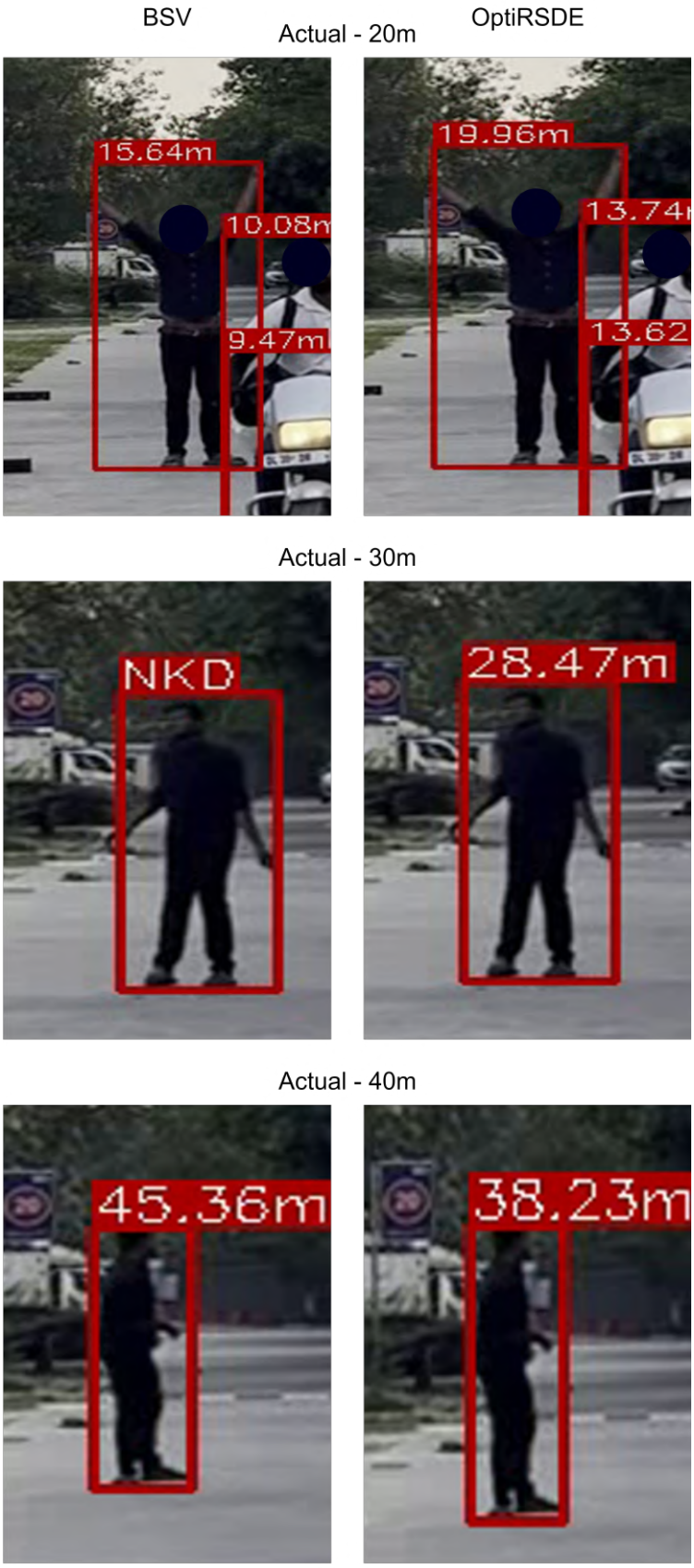

Figure 9: Result 2.

Figure 10: Result 3.

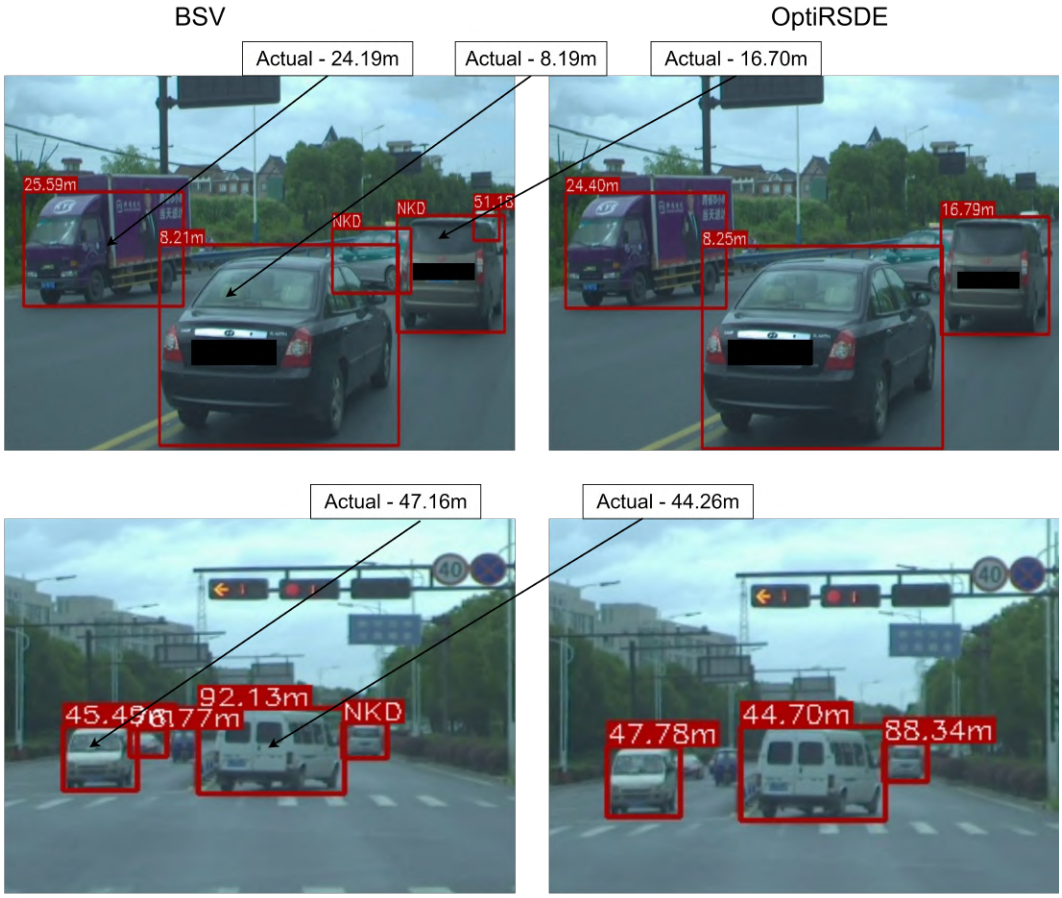

Figure 11: Result 4.