Pak. j. life soc. Sci. (2025), 23(2): 1164-1172 E-ISSN: 2221-7630;P-ISSN: 1727-4915

Pakistan Journal of Life and Social Sciences

www.pjlss.edu.pk

https://doi.org/10.57239/PJLSS-2025-23.2.00244

RESEARCH ARTICLE

Depth Image Reconstruction for Enhanced Slam Accuracy in Agricultural Robot Navigation

N. Gapon[1], V. Voronin[2], A. Zelensky[2], M. Zhdanova[1], E. Semenishchev[3]

[1]Don State Technical University, Rostov-on-Don, Russian Federation
[2]Scientific-Manufacturing Complex «Technological Centre», Zelenograd, Russian Federation
[3]Moscow State University of Technology «STANKIN», Moscow, Russian Federation

ARTICLE INFO

Received: Oct 16, 2025

Accepted: Dec 10, 2025

Keywords

Agricultural Robot
Slam
Depth Map
Neural Network
Anisotropic Gradient
Depth Map Reconstruction

***Corresponding Author:**

narong@pi.ac.th

ABSTRACT

This research aims to enhance the accuracy of autonomous positioning for agricultural robots by developing a novel depth image reconstruction method. This method addresses the issue of data loss in depth images, thereby improving the performance of the Simultaneous Localization and Mapping (SLAM) system. An original depth image reconstruction method was developed, which includes: hierarchical multi-scale search for similar blocks and a scale-adaptive priority function, anisotropic gradient computation; and fusion of the found blocks using a neural network architecture consisting of an encoder, a fusion layer, and a decoder. The method was tested on the Rosario dataset, which includes complex agricultural scenarios. The depth image reconstruction demonstrated a significant improvement in quality: the average error (RMSE) decreased by 20-30%, while the Peak Signal-to-Noise Ratio (PSNR) and the Structural Similarity Index (SSIM) increased by 20-30% compared to existing methods. It is shown that the proposed method preserves the structure and texture of the restored areas, ensuring accurate reconstruction of large zones with missing pixels. To compare SLAM performance, the S-MSCKF algorithm was selected. The quantitative results for Absolute Trajectory Error (ATE) and the mean RMSE were evaluated using the SLAM system before and after the restoration of the depth maps. The Absolute Trajectory Error (ATE) decreased from 0.62 m to 0.25 m, and the RMSE decreased from 0.85 m to 0.39 m. The new method significantly enhances the accuracy of SLAM systems, especially under challenging conditions such as complex rural landscapes, variable lighting, and long-distance travel. The method has the potential for broad implementation in autonomous control systems for agricultural machinery, increasing the reliability and safety of robot operation.

1.INTRODUCTION

Today, robots are used in almost all aspects of our daily lives. They perform various tasks in manufacturing, medicine, science, agriculture, and other fields. Robotics in the agricultural sector [1] has proven its effectiveness for tasks such as seeding [2], fertilizing, precision spraying, irrigation [3], crop monitoring, weed control [4], harvesting, and others. The implementation of the Human-Robot Interaction (HRI) strategy in agriculture promotes process automation and helps to reduce production costs, minimize tedious manual labor, increase the accuracy of mechanized operations, improve the quality of fresh produce, and enhance environmental control. The effective implementation of the HRI concept is achieved by equipping agricultural machinery with Computer Vision Systems (CVS) and the algorithms that power them. CVS can comprise various sensors that enable machines to see the surrounding space, recognize target objects and obstacles, estimate the distance to them, and plan a movement trajectory.

To determine the distance from the CVS sensor to the target object, depth mapping methods are used. A depth map is an image where each pixel contains information about the distance from the camera to the observed object instead of color information [5].

One of the most popular and accessible methods today for building a robot's movement trajectory is SLAM (Simultaneous Localization and Mapping) [6]. It is a method developed to solve the problem of self-localization and mapping in an unknown environment. SLAM is widely used in robot navigation [7], autonomous driving, and augmented and virtual reality. The input data for SLAM algorithms are measurements from sensors used to build the depth map.

One of the problems of SLAM systems is missing areas on the depth map [8], which leads to a decrease in the accuracy of trajectory planning. Such defects occur due to poor lighting, or the presence of reflective or fine-grained surfaces on objects. As a result, an effect of enlarged object (obstacle) boundaries appears, and object occlusion makes it impossible to distinguish one object from another [8].

For robot navigation applications, depth maps created by stereo cameras, RGB-D cameras, and other sensors require additional processing to fill in the missing parts using reconstruction (inpainting) methods. However, traditional color image inpainting techniques cannot be directly applied to depth maps, as there is insufficient information to make accurate inferences about the scene structure. This paper proposes a novel reconstruction method for restoring lost areas in depth maps to enhance the accuracy of autonomous navigation for agricultural robotic platforms.

2.THE PROPOSED METHOD

Autonomous navigation of agricultural robotic systems relies on Simultaneous Localization and Mapping (SLAM) technology [6]. The system uses visual sensors to acquire data required for depth map construction - this is an image $S_{i,j},\ i = 1 \ldots N,\ j = 1 \ldots M$ where the brightness of each pixel corresponds to the distance to the object. Missing regions in the obtained depth images are restored using the proposed image inpainting method (Figure 1).

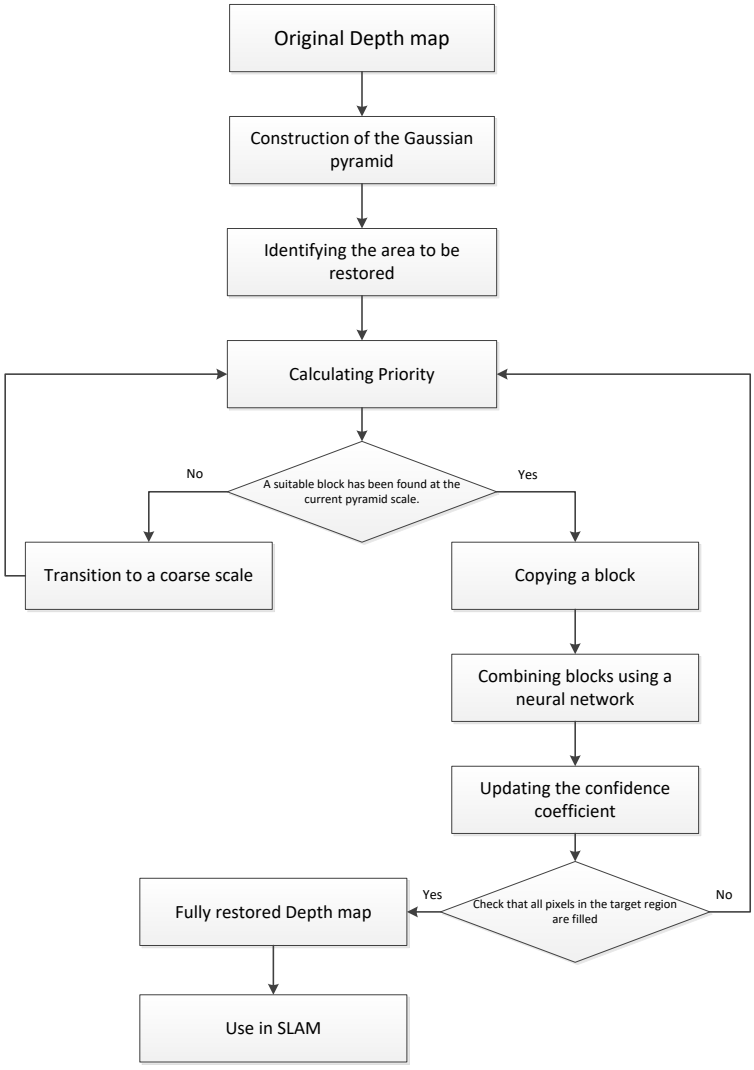

Figure 1 – Block diagram of the proposed method

The method implements block similarity search not only at the original scale but also at multiple reduced resolution levels (via a Gaussian pyramid). This enables:

restoration of global shapes and large structures at low scales (1/4, 1/8);

refinement of textures and details at high scales (1/1, 1/2).

To enhance the robustness and quality of depth map inpainting, particularly under conditions of sparse or damaged data, this paper proposes a modification of the basic image reconstruction method.

The framework consists of two interconnected components: a hierarchical multi-scale search for similar blocks and a scale-adaptive priority function.

Let I be the source image (e.g., RGB or depth); $L \in \{0,1,2,\dots,L_{max}\}$ denote the pyramid levels (where $L = 0$ corresponds to the original resolution, $L = 1$ to $1/2$ resolution, $L = 2$ to $1/4$ resolution, and so on); $I^{(L)}$ represent the image at level L, defined as $I^{(L)} = \text{Downsample}(I, 2^L)$; $\Omega \subseteq \mathbb{Z}^2$ be the target region (defect area); $B_{i,j}^{(L)}$ denote a block of size $s_L \times s_L$ centered at coordinates (i, j) at level L.

Unlike classical similarity search algorithms that operate at a fixed resolution, the proposed approach utilizes a Gaussian pyramid representation of the image $I^{(L)}$, where each level $L \in \{0,1,2,\dots,L_{max}\}$ corresponds to a reduction in spatial resolution by a factor of 2^L:

$$2^L I^{(L)} = \text{Downsample}(I, 2^L).$$

For efficient management of the defect region filling order, a modified priority function dependent on the scale pyramid level is introduced. The priority of a pixel p at level L is defined as:

$$P^{(L)}(p) = C^{(L)}(p) \cdot D^{(L)}(p)$$

where $C^{(L)}(p)$ is the confidence term for the block at scale L, and $D^{(L)}(p) = \left|\nabla^{(L)} I(p) \cdot \vec{n}(p)\right|$ is the gradient magnitude along the normal direction.

The overall priority is determined by considering the reliability level:

$$P(p) = \max_L \left[\lambda^{(L)} \cdot P^{(L)}(p) \cdot \mathbb{1}_{\text{match}(L)}(p)\right]$$

where: $\lambda^{(L)} \in (0,1]$ is the scale weight (e.g., $\lambda^{(L)} = \frac{1}{2^L}$); $\mathbb{1}_{\text{match}(L)}(p) = 1$ if a valid analogous block is found at level L (with an error below the threshold); If no high-quality match is found at $L = 0$, the chance of filling from $L = 1$ and higher levels increases.

The priority calculation focuses on boundary pixels with sharp brightness variations, which contain the maximum amount of information. The confidence coefficient suppresses the influence of already processed pixels, thereby expanding the search area for missing data. The gradient is represented as a vector field where each pixel is characterized by a vector indicating the direction of the greatest intensity change. A novel method for calculating the anisotropic gradient is proposed, ensuring robustness to noise and fine texture details.

To compute shape-dependent patterns, the image is divided into $m \times n$ windows [9]. Features corresponding to the nine structural patterns (Figure 2) are extracted from each window.

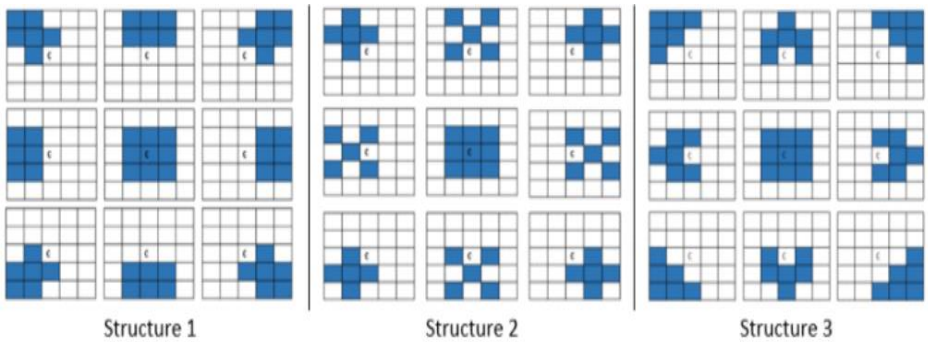

Figure 2 – Structural areas (Structure 1, Structure 2, Structure 3)

Let us calculate the average value of pixel intensity in each region. Let M_c be the average value of the central region, and let M_i be the average value of the i-th directed region (for $i = 1..8$) M_c [10-11](Figure 3).

Central region mean:

$$M_c(x,y) = \frac{1}{|W_c|} \sum_{(u,v)\in W_c(x,y)} I(u,v)$$

where $W_c(x,y)$ is the set of coordinates in the central neighborhood of pixel (x,y).

Directional region mean for direction i:

$$M_i(x,y) = \frac{1}{|W_i|} \sum_{(u,v)\in W_i(x,y)} I(u,v)$$

where $W_i(x,y)$ is the set of pixels in the i-th directional sector around (x,y) as defined by the template.

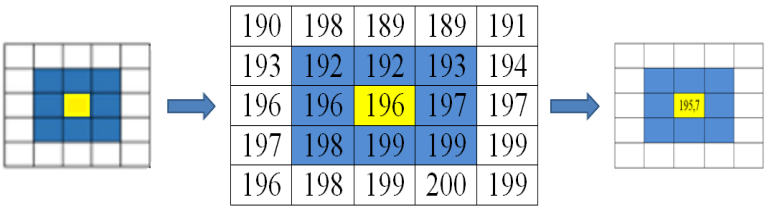

Figure 3 – Example of shape-dependent average calculation

For each direction i, we calculate the gradient as the difference between the mean intensity in that direction and the mean intensity at the center. For example, if $I(x,y)$ is the intensity at the center (Figure 4):

$$G_i(x,y) = M_i(x,y) - M_c(x,y)$$

which measures how much brighter or darker the i-th direction is compared to the center. This is done for all eight directions (covering 0°, 45°, 90°, ..., 315°) around the pixel. All eight directional gradients $\{G_1..G_8\}$ are thus obtained for the current template structure.

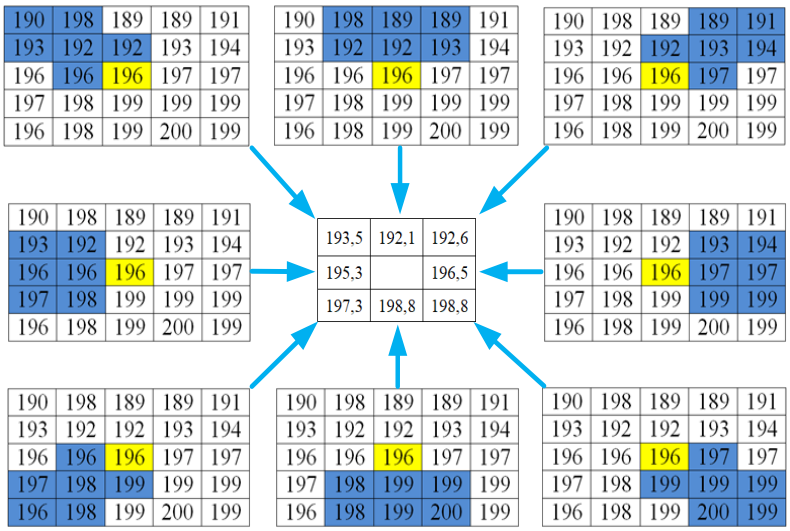

Figure 4 – Example of directional gradient calculation

The above steps (partitioning and gradient computation) are repeated for three different structural templates (referred to as Structure 1, Structure 2, and Structure 3). Each template defines a unique shape configuration for the central and directional regions. In other words, the neighborhood is partitioned in three distinct ways, yielding three sets of directional gradients $G_i\{(1)\}, G_i\{(2)\}, G_i\{(3)\}$ for $i = 1..8$.

Combine the gradient estimates from all three structures to produce a final, robust gradient measurement. This is done by averaging the gradients from the three templates for each direction:

$$G_i^{avg}(x,y) = \frac{1}{3}\left(G_i^{(1)}(x,y) + G_i^{(2)}(x,y) + G_i^{(3)}(x,y)\right), \quad i = 1,\dots,8$$

Averaging the three sets of gradients reduces random noise, since noise tends to affect each structure's measurement differently and thus cancels out when averaged. The final gradient magnitude at pixel (x,y) can be defined from the averaged directional components. For example, one can take the maximum absolute directional gradient as the edge strength:

$$G_{final}(x,y) = \max_{i=1,\dots,8} \left|G_i^{avg}(x,y)\right|$$

Thus, if a region cannot be reliably reconstructed at high resolution, the algorithm automatically switches to a coarser scale to ensure shape recovery. Subsequently, texture details can be refined using data from higher pyramid levels.

At each pyramid level, a search is performed for blocks $B^{(L)}i,j$ in the vicinity of the defective region boundary, which is defined by a binary mask (Figure 5). The similarity metric between blocks is based on the gradient structure:

$$\mathrm{Sim}(B^{(L)}i,j,B^{(L)}m,n) = \frac{1}{s_L^2}\sum u,v\left(\nabla B^{(L)}i,j(u,v) - \nabla B^{(L)}m,n(u,v)\right)^2,$$

where ∇B is the directed anisotropic gradient.

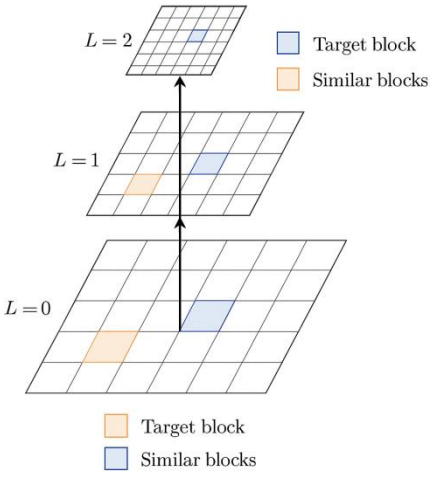

Figure 5 – Visualization of the multi-scale processing scheme

For each level, the optimal block is selected:

$$(\hat{m}L, \hat{n}L) = \arg\min(m,n) \in \mathcal{S}^{(L)}\,\mathrm{Sim}(B^{(L)}i,j,B_{m,n}^{(L)}).$$

The results obtained at different scales are aggregated using a weighted neural decoder, ensuring robust reconstruction of both global contours and local textures. The neural network architecture consists of three components: an encoder, a fusion layer, and a decoder (Figure 6).

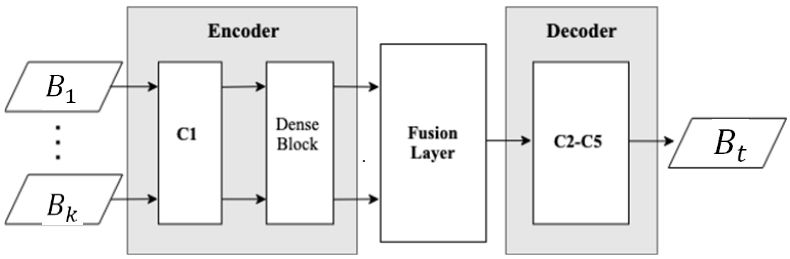

Figure 6 – Architecture of the neural network for combining the found blocks

The input signals consist of the identified blocks, denoted as $B_1, \dots, B_k$. where $k \geq 2$. To merge the blocks obtained from different pyramid levels, a neural autoencoder architecture with parametrized logarithmic feature fusion (PLIP) is proposed. This approach integrates global structural information (available at coarse scales) with detailed texture (from fine scales), ensuring high-quality reconstruction.

Let the target block B_t be the one to be reconstructed. For this block, the most similar blocks $\{B_k^{(L)}\}$. were identified at each pyramid level $L \in \{0,1, \dots, L_{max}\}$. All of them are scaled to a unified resolution (typically $L = 0$ and fed into the neural network:

$$\mathcal{X} = \left\{\mathcal{U}psample\left(B^{(L)}k\right), \quad L = 0,1, \dots, L_{max}\right\}$$

Encoder: Extracts multi-level features $\{F^{(L)}\}$ from each block $B_k^{(L)}$ using convolutional layers and dense blocks. Fusion layer: Employs a PLIP operator or soft-max + PLIP to adaptively combine features extracted from different scales into a unified representation:

$$F_{\text{fused}} = \text{PLIP} - \text{SoftMax}\left(F^{(0)}, F^{(1)}, \dots, F^{(L_{max})}\right)$$

Decoder: Reconstructs the block $\hat{B}_t$ from the merged feature space, ensuring maximal consistency with the image context.

According to the parametrized logarithmic model, the fusion of feature maps $F^{(L)}$ at channel c and position (i, j) is defined as:

$$\tilde{F}c(i,j) = \sum L = 0^{L_{max}} w^{(L)}c(i,j) \odot \text{PLIP}F_c^{(L)}(i,j)$$

where $w_c^{(L)}(i,j)$ is the feature weight from scale L, determined by softmax activation based on confidence level; $\odot_{\text{PLIP}}$ denotes parametrized logarithmic addition; The resulting feature map $\tilde{F}_c$ is fed into the decoder.

The confidence level for a block at scale L is incorporated into the softmax weight assignment:

$$w_c^{(L)}(i,j) = \frac{\exp(a \cdot q^{(L)}c(i,j))}{\sum L' \exp(a \cdot q_c^{(L')}(i,j))}$$

where $q_c^{(L)}(i,j)$ is the quality measure (e.g., local gradient magnitude or block similarity metric); α is a scaling parameter.

This approach leverages global shape features from coarse scales while refining local textures using detailed features from high resolution. Adaptive fusion is implemented based on the confidence level for each scale through a logarithmic model, which helps eliminate artifacts and unreliable information.

After a block is filled with new pixel values, the confidence coefficient $C^{(L)}(p)$ is updated.

This rule simplifies the reliability assessment at the boundaries of the processed region. As the filled area expands, the data reliability decreases, reflecting reduced confidence in pixel color values closer to the center of the target region.

3. RESULTS OF THE EXPERIMENT

The experiment was conducted using the Rosario dataset [12]. Figure 7 presents the depth map reconstruction results: original RGB images (first column), raw depth maps (second column), and depth maps reconstructed using the proposed algorithm (third column).

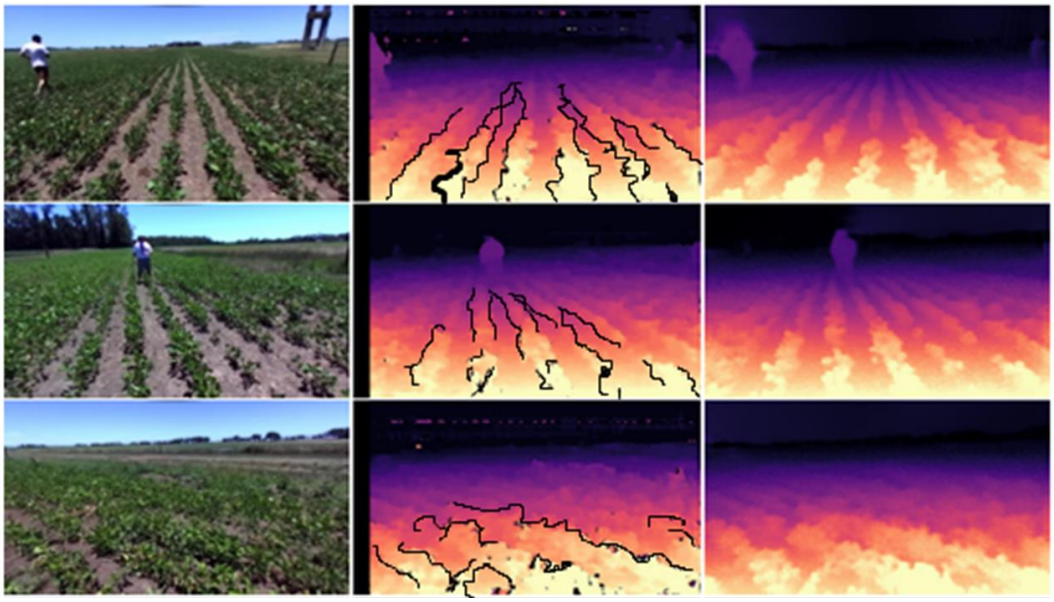

Figure 7 – Results of depth map inpainting from the Rosario dataset

The proposed method ensures accurate boundary restoration without loss of texture and structural clarity, even when reconstructing large areas with missing pixels. Table 1 compares the errors of the proposed method with the EBM [5], DeepFill [13], and EdgeConnect [13] methods on test depth maps from the Rosario dataset.

Table 1 – Error values for the proposed method, EBM, DeepFill and Edge Connect methods

Images	*PSNR*				*RMSE*				*SSIM*			
	EBM,	*DeepFill*	*EdgeConnect*	*Proposed method*	*EBM,*	*DeepFill*	*EdgeConnect*	*Proposed method*	*EBM,*	*DeepFill*	*EdgeConnect*	*Proposed method*
1	16,36	17,05	17,32	18,60	31,19	29,60	27,49	24,68	0,95	0,96	0,97	0,98
2	17,26	18,26	18,45	19,94	27,11	26,20	24,90	20,13	0,95	0,98	0,98	0,98
3	12,23	13,76	14,03	16,53	36,57	35,88	34,01	31,86	0,96	0,96	0,97	0,99
Avg. Dat-se	15,22	16,23	16,54	*17,57*	31,62	30,9	29,08	*26,11*	0,95	0,96	0,98	*0,99*

For the comparative analysis of SLAM algorithm accuracy, S-MSCKF [14] was selected as the benchmark. Table 2 presents quantitative results, showing the mean values of the Absolute Trajectory Error (ATE) and Root Mean Square Error (RMSE) [15] before and after depth image reconstruction. The ATE and RMSE values were computed by averaging the results of five independent runs for each sequence. The best results (lowest errors) are highlighted in bold.

Table 2 – Average RMSE and absolute trajectory error (ATE)

Subsequence	**S-MSCKF before depth map inpainting**		**S-MSCKF after depth maps inpainting**	
	ATE(м)	***RMSE***	***ATE(м)***	***RMSE***
Rosario 01	0,62	0,85	0,25	0,39
Rosario 02	1,25	1,31	0,63	0,74
Rosario 03	1,2	1,33	0,61	0,76

As can be seen from Table 2, the RMSE and ATE values after depth map reconstruction are superior to the RMSE and ATE values before reconstruction.

CONCLUSIONS

An original depth image reconstruction method has been developed, which includes: hierarchical multi-scale search for similar blocks and a scale-adaptive priority function, anisotropic gradient computation; and fusion of the identified blocks using a neural network architecture consisting of an encoder, a fusion layer, and a decoder. The method was tested on the Rosario dataset. The proposed method achieves a 20-30% lower reconstruction error compared to traditional approaches. Its effectiveness has been confirmed through depth map reconstruction experiments.

Challenging conditions, such as repetitive rural landscapes, varying lighting, long trajectories, and dynamic wind effects, pose significant difficulties for state-of-the-art visual SLAM systems. However, the results obtained using S-MSCKF demonstrated the best performance after depth map reconstruction. This opens prospects for applying Visual SLAM in low-cost sensor systems, enhancing the reliability and accuracy of autonomous navigation in agricultural applications.

Acknowledgment

The Scientific Research was funded by the Ministry of Science and Higher Education of the Russian Federation under the Grant «Development of intelligent control methods for technological equipment using the example of semiconductor crystal bonding, and its automation through a machine vision system» (FSFS-2025-0009).

REFERENCES

[1] Klychova G. S., Zakirova A. R., Valiev A. R. [et al.]. Improving the efficiency of crop management systems based on digital technologies // Bulletin of the Kazan State Agrarian University. Vol. 16, No. 3(63). P. 121-127 (2021).

[2] Smirnov I. G., Khort D. O., Kutyrev A. I. Intelligent technologies and robotic machines for cultivating horticultural crops // Agricultural machinery and technologies. Vol. 15, No. 4. P. 35-41 (2021).

[3] Fedorenko V. F., Kharitonov M. P., Smirnov I. G., Aristov E. G. Prospects for robotization of subsurface irrigation and plant feeding processes // Agroengineering. Vol. 26, No. 1. Pp. 11-17 (2024).

[4] Smirnov, I. G., Dyshekov A. I., Devyatkin F. V. Algorithm of operation of an autonomous robotic complex for monitoring weeds // Electrical technologies and electrical equipment in the agro-industrial complex. Vol. 71, No. 1(54). P. 71-75 (2024).

[5] Zelensky A. A., Gapon N. V., Zhdanova M. M., Voronin V. V., Ilyukhin Yu. V. Method for restoring the depth map in problems of robot and mechatronic systems control // Mechatronics, automation, control. Vol. 23, No. 2. P. 104-112 (2022).

[6] Pavlov A. S. Methodology for planning the trajectory of a group of mobile robots in an unknown closed environment with obstacles // Control, Communication and Security Systems. No. 3. pp. 38-59 (2021).

[7] Teterev, A. V. Justification for the choice of a positioning system for controlling the movement of a mobile agricultural robot // Agricultural Machinery and Technologies. Vol. 14, No. 4. pp. 63-70 (2020).

[8] Kutyrev, A. I., Dyshekov A. I. Development of a motion control system for a robotic platform based on laser ranging methods (LiDAR) // Agroinzheneriya. Vol. 25, No. 2. Pp. 19-27 (2023).

[9] Panetta K., Sanghavi F., Agaian S., Madan N. Automated detection of COVID-19 cases on radiographs using shape-dependent Fibonacci-p patterns // IEEE Journal of Biomedical and Health Informatics. Vol.25. No. 6. 1852-1863 (2021).

[10] Zelensky, A., Voronin, V., Semenishchev, E., Gapon, N., Zhdanova, M., Naumov, I.. Medical image segmentation via shape-dependent anisotropic gradient. In Multimodal Image Exploitation and Learning 2025, Vol. 13457, pp. 237-244 (2025)

[11] Zelensky, A., Gapon, N., Zhdanova, M., Voronin, V., Ilukhin, Y., Khamidullin, I. Image inpainting by anisotropic gradient estimation. In Optoelectronic Imaging and Multimedia Technology XI, Vol. 13239, pp. 487-495 (2024).

[12] Podtikhov, A. V., Saveliev A. I. Open dataset for testing Visual SLAM algorithms under various weather conditions // Proceedings of educational institutions of communication. Vol. 10, No. 1. Pp. 97-106 (2024).

[13] Kim H., Kim C., Kim H., Cho S., Hwang E. Panoptic blind image inpainting // ISA transactions. Vol. 132. 208-221 (2023).
[14] Zhang Z., Dong P., Wan, J., Sun Y. Improving S-MSCKF with variational Bayesian adaptive nonlinear filter // IEEE Sensors Journal, Vol.20. No.16. 9437-9448 (2020).
[15] Bokovoy, A. V., Muravyov K. F., Yakovlev K. S. System for simultaneous mapping, localization and exploration of unknown terrain based on video stream // Information technologies and computing systems. No. 2. pp. 51-61 (2020).