



Article Heuristic EPnP-Based Pose Estimation for Underground Machine Tracking

Lingling Su¹, Xianhua Zheng², Yongshi Song², Ge Liu², Nana Chen², Shang Feng³ and Lin Zhang^{2,*}

- ¹ College of Science, North China University of Technology, Beijing 100144, China; sll506@bit.edu.cn
- ² School of Robot Engineering, Yangtze Normal University, Chongqing 408100, China; xianhuazheng@yznu.edu.cn (X.Z.); yongshisong@yznu.edu.cn (Y.S.); g.liu@yznu.edu.cn (G.L.); nanachen@yznu.edu.cn (N.C.)
- ³ Faculty of Computing, Harbin Institute of Technology, Harbin 150001, China; fengshang@hit.edu.cn
- * Correspondence: lin.zhang_2014@hotmail.com or linzhang@yznu.edu.cn

Abstract: Pose estimation is one of the most complicated and compromising problems for underground mining machine tracking, and it is particularly important for hydraulic support autonomous following mining machine (AFM) policy-making system. In this paper, a low-cost infrared visionbased system through an Efficient Perspective-n-Point (EPnP) algorithm is proposed. To improve efficiency and simplify computation, a traditional EPnP algorithm is modified through a natureinspired heuristic optimization algorithm. The optimized algorithm is integrated into the AFM policy-making system to estimate the relative pose (R-Pose) estimation between hydraulic support and the mining machine's shearer drum. Simple yet effective numerical simulations and industrial experiments were carried out to validate the proposed method. The pose estimation error was $\leq 1\%$ under normal lighting and illuminance, and $\leq 2\%$ in a simulated underground environment, which was accurate enough to meet the needs of practical applications. Both numerical simulation and industrial experiment proved the superiority of the approach.

Keywords: fully mechanized underground working face; pose estimation; EPnP; optimization; computer vision

1. Introduction

The pose estimation of intelligent equipment on a fully mechanized underground working face is particularly complicated and significant for unmanned mining. One of the most challenging problems is hydraulic support autonomous following mining machine (AFM), which is aimed at mining machine tracking. The relative pose (R-Pose) between hydraulic support and the mining machine's shearer drum is the most important parameter for ensuring AFM safety and efficiency. The effective monitoring of the R-Pose using the traditional approach, based on inclinometer or computational estimation, cannot give a global view that considers coal-seam geological conditions and hardness indeterminacy, nor can it indicate equipment straightness within a limited monitoring space. Consequently, a major production accident is inevitable.

In the present study, we propose an ArUco-based Perspective-n-Point (PnP) solution for R-Pose estimation and to establish an AFM policy-making system that integrates R-Pose estimation. To improve accuracy and efficiency and simplify computation, the PnP solution is optimized by the nature-inspired heuristic optimization algorithm virus colony search (VCS). The main novelty of the work is the proposed VCS-optimized Efficient Perspective-n-Point (EPnP) algorithm, which was first applied in underground mining machine tracking. The organization of the paper is as follows: In Section 2, related works about visionbased pose estimation, PnP problem and the heuristic algorithm for PnP solution are reviewed. Section 3, the theoretical foundations of vision-based pose estimation and the VCS algorithm are presented. In Section 4, the methodology used for the whole detection



Citation: Su, L.; Zheng, X.; Song, Y.; Liu, G.; Chen, N.; Feng, S.; Zhang, L. Heuristic EPnP-Based Pose Estimation for Underground Machine Tracking. *Symmetry* **2022**, *14*, 385. https://doi.org/10.3390/ sym14020385

Academic Editor: Basil Papadopoulos

Received: 15 January 2022 Accepted: 11 February 2022 Published: 15 February 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/).

2 of 17

procedure is demonstrated. In Section 5, key technologies, including the system architecture, AFM policy-making flowchart and the VCS-optimized E-PnP algorithm are described. In Section 6, numerical simulation is conducted and industrial experiments carried out with physical equipment. Conclusions are summarized in Section 7.

2. Literature Review

2.1. Vision-Based Pose Estimation

In essence, the proposed AFM policy-making system uses computer vision to estimate the R-Pose. At present, vision-based pose estimation is widely used in aircraft visual inspection [1], autonomous vehicles [2], civil construction [3] and human body pose recognition [4]. Concerning robotic grasping, Guoguang Du et al. [5], conducted a comprehensive survey that reviewed three key tasks during vision-based robotic grasping: object localization, object pose estimation and grasp estimation. Due to the characteristics of texturelessness and self-occlusion, Jia [6] proposed a machine-vision-based method for 6D pipe pose estimation. By combing vision and tactile sensors, Dan et al. [7] proposed a novel, accurate positioning method for object pose estimation in robotic manipulation. Praneet et al. [8] developed OpenMonkeyStudio, a deep-learning-based markerless motion capture system to estimate a 3D pose in freely moving macaques in large unconstrained environments. To quantitatively analyze impaired movement from neurological and musculoskeletal diseases, Łukasz Kidziński [9] proposed a deep neural network to predict clinically relevant motion parameters from an ordinary video of a patient. Valliappan [10] also used vision-based technology for eye-tracking, which uses machine learning to demonstrate accurate smartphone-based eye tracking without any additional hardware. Vision-based pose estimation has also been applied in aerospace. For example, Thaweerath [11] et al. proposed a deep convolutional neural network for noncooperative docking operations through vision-based spacecraft pose estimation. Sukkeun Kim et al. [12], proposed a kind of vision-based pose estimation for fixed-wing aircraft using one-short looking and PnP.

2.2. PnP Problem

The Perspective-n-Point (PnP) pose problem concerns the estimation of the relative pose between a calibrated camera and an object from a set of n 3D coordinates points (x,y,z) and their 2D projections with known (u,v) pixel coordinates. As early in 1989, Radu et al. [13] provided an analytic solution to the Perspective 4-Point problem. The most classical was proposed by Moreno [14] in 2007, a non-iterative solution that grew linearly and had limited computational complexity. Then, Li et al. [15] proposed a noniterative solution that could robustly retrieve the optimum by solving a seventh-order polynomial. Wang Ping [16] proposed a simple, robust and fast algorithm that translated the pose estimation problem into an optimal problem requiring only a seventh-order and fourth-order univariate polynomial to be solved. It made the processes more easily understandable and significantly improved performance. Recently, other novel solutions have been proposed. For example, Zhou [17] proposed an efficient PnP solution using an uncalibrated camera of unknown focal length. Yu [18] presented an efficient algebraic solution to the perspective 3-point problem in which the camera pose was estimated from three given 3D–2D correspondence sets. Zhou [19], however, estimated the camera pose using $n \ge 3$ 2D–3D line correspondences. Thus, the problem was turned into a Perspectiven-Line problem. Finally, a complete, accurate, and efficient solution was proposed by solving the minimal (n = 3) problem and the least-squares problem (n > 3) in different ways. Particularly noteworthy are the new approaches via machine learning that were studied as well. Liu et al. [20] proposed a deep convolutional neural network (CNN) model that simultaneously solves for both the six degrees of freedom (6-DoF) absolute camera pose and the 2D–3D correspondences. More intelligent and heuristic PnP solutions need further discussion.

2.3. Heuristic Optimization

Based on the research trend of vision-based pose estimation, optimization is the key to intelligent and heuristic PnP solutions. Recently, nature-inspired optimization has been widely used in image processing [21], text document clustering [22], industrial data mining [23] and decision-making problems [24]. The algorithms most commonly used in computer vision are ant colony [25], particle swarm [26], bee colony [27], Grey Wolf [28], and Cockroach Colony [29]. In 2019, Mohit et al. [30] proposed a novel nature-inspired algorithm: Squirrel search. The typical optimization benchmark functions were tested, and six well-known optimization algorithms were compared to it. The results showed the superiority of the proposed algorithm. Then, Ghasemi-Marzbali [31] presented a novel nature-inspired meta-heuristic optimization; bear smell search algorithm (BSSA) that took into account powerful global and local search operators. Neetesh [32] demonstrated the dynamic foraging behavior of Agama lizards and built a mathematical model to simulate their foraging methods as an the artificial lizard search optimization (ALSO) algorithm. Among these approaches, virus colony search (VCS) [33] is a novel naturespired algorithm that simulates diffusion and infection strategies for virus-infected host cells to survive and propagate in the cell environment. Because it considers convergence and accuracy simultaneously, the VCS is particularly suitable for global numerical and engineering optimization problems. Thus, in this paper, the VCS algorithm was applied to PnP optimization, and a heuristic EPnP-based pose estimation approach for hydraulic support autonomous followed.

3. Theoretical Foundations

The key to vision-based R-Pose estimation is the PnP solution. To improve estimation accuracy, an R-Pose estimation method based on the VCS-optimized EPnP algorithm was proposed, which is of great significance for AFM policymaking.

3.1. EPnP-Based Pose Estimation

The essence of the R-Pose estimation is to obtain the relative pose between the camera and ArUco maker through *n* 3D points (x, y, z) and their 2D projection coordinates (μ , v). To solve the PnP problem, Lepetit et al. [14] proposed an efficient PnP (EPnP) algorithm with a time complexity of only O(n). During the solution, four coordinate systems were established: ArUco maker, imaging plane, camera and world. According to the small-hole imaging principle, the conversion equation between the camera coordinate and world coordinate can be calculated as follows:

$$Z_{C}\begin{bmatrix} u\\v\\1\end{bmatrix} = \begin{bmatrix} f_{u} & 0 & u_{0} & 0\\0 & f_{v} & v_{0} & 0\\0 & 0 & 1 & 0\end{bmatrix} \begin{bmatrix} R & T\\O & 1\end{bmatrix} \begin{vmatrix} X_{w}\\Y_{w}\\Z_{w}\\1\end{vmatrix}$$
(1)

where *R* is the 3 × 3 rotation matrix; *T* is the 3 × 1 displacement vector; $f_u = f/dx$; $f_v = f/dy$; and *f* is the focal length of camera. The first matrix on the right of the above formula is the internal parameter matrix. The second is the rigid body transformation matrix, i.e., the external parameter matrix. $[u v 1]^T$ is the projective pixel point on the marker image and $[X_w Y_w Z_w]^T$ is the 3D coordinate of reference point on the marker in the world coordinate. To solve *R* and *T*, four control points $c_B^j(j = 1, 2, 3, 4)$ were introduced to express the coordinates of the reference point on the marker. The coordinates of the reference and control points can be converted as follows:

$$p_B^i = \sum_{j=1}^4 a_{ij} c_B^j, i = 1, 2, \dots, N$$
 (2)

where p_B^l is the reference point on the marker, and the vector a_{ij} is the coordinate of the spatial reference point in Euclidean space. According to the invariance of the linear relationship under a Euclidean transformation, the relationship can be calculated as follows:

$$p_C^i = \sum_{j=1}^4 a_{ij} c_C^j, i = 1, 2, \dots, N$$
 (3)

Furthermore, we get

$$k^{i}q^{i} = \sum_{j=1}^{4} a_{ij}c_{C}^{j}, i = 1, 2, \dots, N$$
(4)

It is converted into a system of linear equations with the following formula:

$$Mx = 0, x = \begin{bmatrix} (c_{C}^{1})^{T} & (c_{C}^{2})^{T} & (c_{C}^{3})^{T} & (c_{C}^{4})^{T} \end{bmatrix}$$
(5)

The above equations have 4 control points and 12 unknown variables, and x is the right zero space of M. The solution of x is the kernel space of M, which can be obtained by solving the zero-space characteristic sequence for $M^T M$. x can be obtained by choosing the linear combination sequence corresponding to the minimum zero-space error. The coordinates of the control points in the camera coordinate are restored. In the process of solving x, it was necessary to select an appropriate linear combination:

$$x = \sum_{i=1}^{N} \beta_i v_i \tag{6}$$

The desirable range of the zero-space dimension for $M^T M$ is N = 1, 2, 3, 4. According to different value of N, there will be different solution strategies. In the computation process of EPnP, the reprojection error (RE) can be calculated corresponding to N is 1, 2, 3 and 4 respectively. Then the result is the one corresponding to the minimum projection error. Suppose the optimization variable is $\beta = [\beta_1, \beta_2, \beta_3, \beta_4]$, then the optimization objectives are set to the following equation.

$$f(\beta) = \sum_{(i,j)s.t.i < j} \left(\left\| c_C^i - c_C^j \right\|^2 - \left\| c_B^i - c_B^j \right\|^2 \right)$$
(7)

where $\|c_B^i - c_B^j\|^2$ is the distance in the object-coordinate system. The coordinates of the control point under the camera coordinate are shown as follows:

$$c_C^i = \sum_{j=1}^4 \beta_j v_j^i \tag{8}$$

After obtaining the pose *R* and *T* of the ArUco maker in the camera coordinate through the above calculation, the pose of the shearer drum can be calculated.

3.2. VCS-Based Optimization

The VCS algorithm simulates the process of virus infection and diffusion into host cells in the cellular environment. The process is divided into three stages: virus diffusion, host–cell infection and host–immune response. According to the characteristics of the three stages of virus growth, the VCS algorithm was demonstrated as follows:

Step 1: Initialize the VCS parameters, including the dimension of optimization problem D, search range [LB, UB], virus and host-cell population size N, and maximum number of iterations gMax. The whole variable space is divided into two groups: virus V_{pop} and host H_{pop} .

Step 2: For virus diffusion, each individual virus will randomly generate new individuals. To find the global optimization, the Gaussian random method was used to generate them. The calculation was presented as follows:

$$V_{pop,i}' = Gaussian \left(G_{best}^g \cdot \log(g) / g \cdot \left(V_{pop,i} - G_{best}^g \right) \right) + \left(r_1 G_{best}^g - r_2 V_{pop,i} \right)$$
(9)

where *i* is the index of randomly selected individuals i = [1, 2, 3, ..., N]; *N* is the size of the population; G_{best}^g is the global optimal solution in the *g*th iteration; and r_1 and r_2 are random coefficients within [0, 1].

Step 3: The process of virus infection is essentially an interactive process of material conversion between host cells and the virus, which can be simulated by the CMA-based evolutionary method [34]. The host individual can be updated as follows:

$$H_{pop,i}^{g} = X_{mean}^{g} + \delta_{i}^{g} \cdot N_{i}(0, C^{g})$$
⁽¹⁰⁾

where $N_i(0, C^g)$ is the mean; the covariance matrix is the Gaussian distribution for $C^g \in \Re^{D \times D}$; *D* is the dimension of the problem; δ_i^g is the step size; X_{mean}^g is the parental vector with dimension number $\lambda = \lfloor N/2 \rfloor$; and its initial value is calculated as

$$X_{mean}^{0} = \frac{\sum_{i=1}^{N} V_{pop,i}}{N}$$
(11)

and can be updated as follows:

$$X_{mean}^{g+1} = \ln(\lambda+1) / \left(\sum_{j=1}^{\lambda} \ln(\lambda+1) - \ln(j)\right)$$
(12)

The step size δ_i^g and covariance matrix C^g are updated through Equations (16) and (17), respectively.

$$\delta_i^{g+1} = \delta_i^g \times \exp\left(\frac{c_\sigma}{d_\sigma} \left(\frac{\|p_\sigma^{g+1}\|}{E\|N(0,I)\|}\right)\right) \tag{13}$$

$$C^{g+1} = \xi_1 C^g + c_1 p_c^{g+1} \left(p_c^{g+1} \right)^T + c_\lambda \sum_{i=1}^{\lambda} w_i \frac{V_{pop,i} - X_{mean}^g}{\sigma^g} \cdot \frac{\left(V_{pop,i} - X_{mean}^g \right)^T}{\sigma^g}$$
(14)

where coefficient $\xi_1 = (1 - c_1 - c_\lambda)$. Coefficient c_1 and c_λ are calculated as follows:

$$c_1 = \frac{1}{\lambda_w} \left(\left(1 - \frac{1}{\lambda_w} \right) \min\left\{ 1, \frac{2\lambda_w - 1}{\left(N+2\right)^2 + \lambda_w} \right\} + \frac{1}{\lambda_w} \frac{2}{\left(N+\sqrt{2}\right)^2} \right)$$
(15)

$$\lambda = (\lambda_{\rm w} - 1)c_1 \tag{16}$$

 p_{σ}^{g+1} and p_{c}^{g+1} are evolutionary paths, which can be calculated as follows:

С

$$p_{\sigma}^{g+1} = (1 - c_{\sigma})p_{\sigma}^g + \frac{\sqrt{c_{\sigma}(2 - c_{\sigma})\lambda_w}}{\sigma^g} (C^g)^{-1/2} \left(X_{mean}^{g+1} - X_{mean}^g\right)$$
(17)

$$p_{c}^{g+1} = (1 - c_{c})p_{c}^{g} + \frac{h_{\sigma}\sqrt{c_{c}(2 - c_{c})\lambda_{w}}}{\sigma^{g}}(C^{g})^{-1/2}\left(X_{mean}^{g+1} - X_{mean}^{g}\right)$$
(18)

 $(C^g)^{-1/2}$ is a symmetric positive matrix that satisfies $(C^g)^{-1/2}(C^g)^{-1/2} = (C^g)^{-1}$; the cumulative coefficients are $c_{\sigma} = (\lambda_w + 2)/(N + \lambda_w + 3), c_c = 4/(N + 4)$ and $d_{\sigma} = 1 + c_{\sigma} + 2\max\{0, (\sqrt{\lambda_w - 1}/\sqrt{N + 1})\}$.

Step 4: Host-cell immunity. This process actually screens for viruses and maintains the virus's strong viability. Therefore, it is necessary to evaluate and sort the fitness of the

virus individual and carry out retention and evolution operations on it according to the sorted value. The specific calculation is as follows:

$$\begin{cases}
V''_{pop,i,j} = V_{pop,i,j} & if r > \Pr_{rank(i)} \\
V''_{pop,i,j} = V_{pop,k,j} + rand \cdot \left(V_{pop,h,j} - V_{pop,i,j}\right) & otherwise
\end{cases}$$
(19)

where k, i, h are unequal integers with random selection, and $j \in [1, 2, 3, ..., d]$. The defined boundary value of sorted individual fitness is $Pr_{rank(i)} = (N - i + 1)/N$.

Step 5: If it reaches the specified end condition, the final individual is retained and its fitness value is calculated. If the stop condition is not reached, return to Step 1.2, and continue searching.

4. Methodology

According to Lepetit et al. [14], our methodology is a four-step optimization problem based on an EPnP-based pose estimation. Let the reference points be p_B^j , and the 4 control points used for expressing their world coordinate be c_B^j (j = 1, 2, 3, 4). The coordinates of the reference point and the control point can be converted as Equation (2). According to the invariance of the linear relationship under the Euclidean transformation, the relationship can be calculated as Equation (3). Furthermore, we get

$$k^{i}q^{i} = \sum_{j=1}^{4} a_{ij}c_{C}^{j}, i = 1, 2, \dots, N$$

It is converted into a system of linear equations with the formula

$$Mx = 0, \ x = \begin{bmatrix} (c_{C}^{1})^{T} & (c_{C}^{2})^{T} & (c_{C}^{3})^{T} & (c_{C}^{4})^{T} \end{bmatrix}$$

The above equations have 4 control points and 12 unknown variables. x is the right zero space of M. The solution of x is the kernel space of M, which can be obtained by solving the zero-space characteristic sequence for $M^T M$. x can be obtained by choosing the linear combination sequence corresponding to the minimum zero space error. The coordinates of the control points in the camera coordinate are restored. In the process of solving x, it is necessary to select an appropriate linear combination, which is also presented in Equation (6).

The desirable range of zero space dimension for $M^T M$ is N = 1, 2, 3, 4. According to different value of N, there will be different solution strategies. In the computation process of EPnP, the reprojection error (RE) can be calculated corresponding to n is 1, 2, 3 and 4 respectively. Then the result is the one corresponding to the minimum projection error. Suppose the optimization variable is $\beta = [\beta_1, \beta_2, \beta_3, \beta_4]$, then the optimization objectives are set to be Equation (7). The coordinates of the control point under camera coordinate are calculated by Equation (8). After obtaining the pose R and T of the ArUco maker in the camera coordinate through the above calculation, the pose of the shearer drum can be calculated.

The installation position of ArUco marker is shown in Figure 1. The local coordinates $\{O_B\}$ and $\{O'_B\}$ are set at the center point *P*0 of the ArUco maker and the center point *P*5 of the shearer drum, respectively. For the convenience of calculation, $\{O_B\}$ is installed parallel to the coordinate axis $\{O'_B\}$. Then, the pose of $\{O'_B\}$ can be calculated from the pose of $\{O_B\}$ through the translation transformation $\vec{l'_r}$.



Figure 1. Installation position of the ArUco maker.

By using ArUco library, the estimated pose can be described as a translation vector T_{vec} and a rotation vector R_{vec} , which can be converted into the rotation matrix R:

$$R = \cos(\theta)I + (1 - \cos(\theta))rr^{T} + \sin(\theta) \begin{vmatrix} 0 & -r_{z} & r_{y} \\ r_{z} & 0 & -r_{x} \\ -r_{y} & r_{x} & 0 \end{vmatrix}$$
(20)

where θ is the rotation angle, and r_x , r_y , r_z are the components of the unit vector of the rotation vector on the *x*, *y* and *z* axes. The pose matrix of the ArUco marker is then described as follows:

$$T_p^B = \begin{bmatrix} R & T_{opt} \\ O & 1 \end{bmatrix}$$
(21)

According to the transformation relationship between $\{O_B\}$ and $\{O'_B\}$, the pose matrix at the center point of shearer drum is

$$\Gamma_p^{B'} = \begin{bmatrix} R & T_{opt} + \overrightarrow{l}_r \\ O & 1 \end{bmatrix}$$
(22)

So far, the relative pose detection between the hydraulic support and shearer drum is realized. Then, through the structural parameters of the shearer rocker arm and drum, the current position and cutting interference state of the mining machine was obtained.

5. Key Technologies

5.1. System Architecture

The key of the AFM is to realize the R-Pose perception between the hydraulic support and mining machine as well as to avoid cutting interference between the hydraulic support and shearer drum. In the proposed system, an infrared camera was installed on the top beam of the hydraulic support, and an ArUco marker was installed on the rocker arm of the mining machine. Considering that the ArUco marker may be blocked during actual coal mining, it is appropriately translated along the rocker arm direction. The pose of the shearer drum can be calculated from the pose of the ArUco maker through translation transformation. Image processing and marker recognition are carried out through the local controller. The architecture of the AFM policy-making system is shown in Figure 2.



Figure 2. Architecture of the AFM policy-making system.

5.2. Flowchart of the Proposed Approach

Figure 3 presents the flowchart of the AFM policy-making system, which is aimed at estimating the relative pose between hydraulic support and shearer drum. Since the fully mechanized underground working face environment has high dust and low illumination, the marker image contains a lot of noise. Therefore, it is necessary to denoise the image before marker detection. By identifying each corner of the marker, the 3D–2D correspondence set is obtained, and the problem of R-Pose estimation is transformed into the PnP problem.



Figure 3. The flowchart of the AFM policy-making method.

Firstly, the internal parameters of the camera should be calibrated. The chosen infrared camera provides built-in night vision to overcome the uneven brightness and low illumination. Images are acquired through an Ethernet interface. The marker installed on the rocker arm of the mining machine adopts the standard marker in the ArUco library. According to the method proposed by Garrido-Jurado [35], the marker with any identification can be selected from different types of dictionaries. Each marker contains a unique dictionary ID, and the marker image size is 200×200 (pixels). The physical length of the marker is all you need for recognition.

Then, the calibrated camera is used to obtain the installation position parameters of camera and the installation position parameters of the marker. The ArUco marker image is collected by infrared camera. A series of 3D–2D correspondence sets can be obtained through image segmentation and recognition and the corner detection of ArUco marker. R-Pose estimation can be transformed into a PnP problem using the 3D–2D correspondence set. On this basis, the relative posture of the hydraulic support and the shearer drum can be obtained by combining with the installation position parameters. Both the position of the mining machine and the cutting interference state can be detected.

Finally, according to the position of the mining machine and cutting interference state, the AFM policy can be obtained through further operation, which is beyond the scope of this paper.

5.3. VCS-EPnP Estimation

During the EPnP process, the Gauss Newton method was applied for β optimization, while the selection of an initial value had a great impact on the optimization results. To make it simple and effective, an optimized EPnP algorithm was proposed based on the VCS (EPnP-VCS) to estimate the R-Pose. The VCS is capable of heuristically searching for the best β . The linear optimal combination of β^* corresponding to the minimum reprojection error was solved through numerical iteration. The EPnP-VCS algorithm focuses on searching for an appropriate β^* to obtain the optimal linear combination scheme. The specific process of the EPnP-VCS algorithm is shown in Figure 4. The calculation steps of the algorithm are described as follows:



Figure 4. Procedure of the EPnP-VCS algorithm.

Step 1. Initialize parameters required by the VCS algorithm, including data dimension D, search range [*LB*, *UB*], population size N, and the maximum number of iterations gMax. At the same time, before collecting the infrared image of the ArUco marker, the calibration of the infrared camera was carried out to obtain the internal parameter matrix A.

Step 2. Define the coordinates of control points c_B^j (j = 1, 2, 3, 4); calculate the linear combination coefficient of 3D points; and calculate the kernel of *M* according to Mx = 0.

Step 3. Estimate the corresponding pose R_{est} , T_{est} and calculate the reprojection error in various cases (i.e., N = 1, 2, 3, 4) taking into consideration the linear combination corresponding to the minimum zero-space error of $M^T M$. The results corresponding to the minimum reprojection error in the four combinations are to be compared, and the estimated reference point coordinates, rotation vector R_{est} , translation vector T_{est} and linear combination parameters a_{ij} are used for VCS optimization.

Step 4. Randomly set the initial value of the search variable β ; initialize the VCS parameters; and estimate the pose estimation R_{est} , T_{est} corresponding to β through the heuristic search in the three stages of VCS—diffusion, infection and immunity.

Step 5. Calculate the reprojection error by using R_{est} and T_{est} , the parameter matrix in the infrared camera and the coordinates of the reference point. If the reprojection error meets the requirements or reaches the maximum number of iterations, the current optimal linear combination β^* is stored; Otherwise, return to Step 4.

Step 6. Calculate R_{opt} and T_{opt} using the optimal linear combination β^* and obtain the final solution.

6. Numerical Simulation and Experimental Study

6.1. Numerical Simulation

For numerical simulation and comparison, the EPnP-VCS, EPnP and EPnP-Gaussian are implemented in MATLAB R2019a. Both hardware configuration and software parameters of the simulation are shown in Table 1:

Hardware	Configure	Parameter Name	Numerical Value
Processor	Intel Core i7, 2.5 GHz	Maximum iterations	100
RAM	8 G	Population size	20
Operating system	Win 10, x64	Variable boundary	[-30, 30]
Software tool	MATLAB	Data maintenance	4

Table 1. Hardware configurations and parameters of the VCS algorithm.

The specific simulation steps are as follows:

Firstly, the calibration parameters of the infrared camera are specified. The focal length is f = 1, the pixel length on the *x* axis and *y* axis is $f_x = f_y = 800$, respectively. The image size is 640×480 px and the internal parameter matrix is A = [800, 0, 320; 0, 800, 240; 0, 0, 1]. The simulated spatial position of the infrared camera is shown in Table 2. If the number of the 3D–2D correspondence set is n = 500, then its spatial coordinates are shown in Figure 5a.

Table 2. Simulated camera position.

x	y	z	x	у	z	x	y	z
-1.611	0.398	4.607	0.364	-0.822	8.325	0.473	-1.491	7.145
0.781	1.463	7.491	0.495	-0.380	4.188	1.279	1.540	7.076
-0.524	-0.191	8.039	-1.935	0.253	6.213	1.500	0.530	4.246
-1.722	0.221	8.795	1.663	1.553	8.324	-1.371	-1.369	8.158
-1.073	1.384	4.732	-1.334	0.887	6.399	-1.249	0.418	5.622



Figure 5. Simulated camera position and image data; (**a**) Simulated camera position; (**b**) Simulated image data.

Secondly, the ground-truth image of the simulation experiment is obtained through the projection of infrared-camera coordinates on the imaging plane-coordinate system. To simulate the possible deviation of image acquisition in the real situation, Gaussian noise with variance σ is added to the ground-truth image. The obtained true value and simulated image are shown in Figure 5b.

Then, rotation matrix and translation vector of the infrared camera are set R = [0.727, -0.685, 0.045, 0; 0.686, 0.725, -0.0600, 0; 0.009, 0.074, 0.997, 0; 0, 0, 0, 1]. To establish the 3D–2D correspondence set, the infrared camera is transformed to a specific position by using the rotation matrix and translation vector. Then, a set of 3D points is obtained as the reference point set.

Finally, the EPnP, EPnP-Gaussian and EPnP-VCS were applied to solve the pose of the generated 3D–2D correspondence set. The results are shown in Figures 6–8, respectively. According to the data handling method [36], the errors used in left part are a reprojection error using a calculated rotation matrix and translation vector. The errors used in right part are the differential error of camera's position in x, y, z directions.

As can be seen from Figure 6b, the estimation error of the infrared camera's position in x, y, z directions are too large (±0.15). The deviation on the z axis is particularly obvious, and the mean error of reprojection is 12.411. After optimizing the EPnP through the Gaussian Newton method, the estimation error in x, y, z directions shows improvement. The error has been reduced to within ±0.04. However, as shown in Figure 7, the deviation on the z axis is still large, i.e., the reprojection error is 12.365. Finally, the EPnP-VCS was applied for comparison. The estimation error in x, y, z directions was significantly decreased. As shown in Figure 8, not only was the error is reduced to ±0.03, but the estimation deviation on the z axis was greatly improved, and the reprojection error was only 12.261.



Figure 6. Simulation result of EPnP; (a) Reprojection error of EPnP; (b) RMSE: x = 0.001; y = 0.005; z = 0.05.



Figure 7. Simulation result of EPnP-Gaussian; (a) Reprojection error of EPnP-Gaussian; (b) RMSE: x = 0.002; y = 0.004; z = 0.03.



Figure 8. Simulation result of EPnP-VCS; (a) Reprojection error of EPnP-VCS; (b) RMSE: x = 0.0003; y = 0.004; z = 0.007.

Based on the above analysis, the EPnP-VCS did not need an empirical initial value of β and had high solution accuracy. Meanwhile, it was found that the population number of the VCS had a certain impact on the stability of the EPnP-VCS. Therefore, the population size was set to 20, 50, 100 and 200, respectively, and the above three methods were repeated 100 times. Th simulation results are shown in Figure 9. The errors are the mean square error. It can be seen from the figure that both the EPnP-Gaussian and EPnP-VCS greatly reduced the reprojection error. The EPnP-VCS had more accurate results than the EPnP-Gaussian with increased population.



Figure 9. The reprojection error of repetitive experiment.

Through the statistics of the reprojection error of the repeated simulations compared with the non-optimized EPnP, the EPnP-Gaussian and EPnP-VCS had more accurate solutions in most cases. Moreover, the average minimum reprojection error of the EPnP-VCS was the smallest. To study the algorithm stability of the EPnP-VCS, we establish different 3D–2D correspondence sets, repeated the above EPnP-VCS simulations and counted its variance. The results are shown in Table 3.

EP	EPnP		EPnP-Gaussian		EPnP-VCS	
Mean	Variance	Mean	Variance	Mean	Variance	
12.64772	0.096387	12.50771	0.084461	12.50349	0.091663	
12.59365	0.091691	12.49178	0.085643	12.48746	0.087089	
12.60194	0.075147	12.50164	0.063751	12.49039	0.060059	
12.6094	0.114787	12.49762	0.090848	12.47782	0.059655	
	EP Mean 12.64772 12.59365 12.60194 12.6094	EPnPMeanVariance12.647720.09638712.593650.09169112.601940.07514712.60940.114787	EPnPEPnP-GMeanVarianceMean12.647720.09638712.5077112.593650.09169112.4917812.601940.07514712.5016412.60940.11478712.49762	EPnPEPnP-GaussianMeanVarianceMeanVariance12.647720.09638712.507710.08446112.593650.09169112.491780.08564312.601940.07514712.501640.06375112.60940.11478712.497620.090848	EPnP EPnP-Gaussian EPnP Mean Variance Mean Variance Mean 12.64772 0.096387 12.50771 0.084461 12.50349 12.59365 0.091691 12.49178 0.085643 12.48746 12.60194 0.075147 12.50164 0.063751 12.49039 12.6094 0.114787 12.49762 0.090848 12.47782	

Table 3. Statistical result of reprojection error produced by repetitive simulation.

To validate the uncertainty and reliability further in a more explicit way, the results are demonstrated in Figure 10. When the population of the VCS was small, its overall solution accuracy was still higher than that of the EPnP-Gaussian and EPnP, but the error changed greatly and the stability was insufficient. With the increase of population, its stability gradually improved, but the computational time also increased accordingly. In practical application, the computational time can be reduced by improving the calculation capacity of equipment and using parallel operation to meet the application requirements of real-time monitoring in a fully mechanized mining face.



Figure 10. Uncertainty and reliability validation by repetitive simulation.

6.2. Industrial Experiments

In this experiment, the image of the ArUco marker was collected by an infrared camera. The image was processed by the denoising method proposed in our previous study [37]. The denoised image was used for R-Pose estimation by EPnP-VCS. To verify the estimation error, a lidar sensor, M16 of Leddar Tech, was used to measure the true distance between the infrared camera and ArUco marker. The M16 sensor and its installation position are shown in Figure 11.





Figure 11. Sensor installation for AFM policy-making experiment.

As shown in Figure 12a–c, three R-Pose statuses are demonstrated under normal lighting conditions. The ground-truth distances between the marker and the infrared camera are 1217, 1228 and 1248 mm, respectively. The recognized reference coordinate and corner positions are shown in Figure 12d–f. By the project coordinates of each corner in the imaging plane-coordinate system, the 3D–2D correspondence set was obtained.



Figure 12. Three R-Pose statuses under normal lighting and illuminance.

Then, through the EPnP-VCS, the 3D–2D correspondence set was used for pose estimation to obtain the rotation vector R_{vec} and translation vector T_{vec} between the infrared camera and ArUco marker. To analyze the error with real distance measured by M16, the spatial distance was calculated by R_{vec} and T_{vec} , and the error generated by the final pose estimation is shown in Table 4.

Pose State	True Value (/cm)	Estimated (/cm)	Error (%)
1	130	131.25	0.96
2	125	125.83	0.66
3	123	122.89	0.09

Table 4. Pose estimation errors under normal lighting and illuminance.

The maximum error between the measured distance estimated by vision and the true distance measured by M16 was no more than 1%, which easily met the accuracy requirements in the fully mechanized mining face. In addition, in this experiment the uneven illumination at night was used to simulate the underground environment of a fully mechanized mining face with poor illumination. The original images collected at night are shown in Figure 13a–c. The corner detection results are shown in Figure 13d–f.



(d)Estimated D=130.29 cm

(e)Estimated D=126.84 cm

(f)Estimated D=123.56 cm

Figure 13. Three relative pose status under simulated underground environment.

Finally, the error between the measured and estimated distances was obtained by the EPnP-VCS, as shown in Table 5. There was little correlation between the estimation errors and those under normal illumination (no more than 2%), so the scheme is feasible.

Table 5. Pose	estimation e	errors under	simulated	underground	environment

Pose State	True Value (/cm)	Estimated (/cm)	Error (%)
1	130	130.29	0.22
2	125	126.84	1.47
3	123	123.56	0.46

7. Conclusions

In this paper, an ArUco-based PnP solution for R-Pose estimation was proposed and an AFM policy-making system was established with the integration of R-Pose estimation. To improve both efficiency and accuracy and to simplify computation, the PnP solution was optimized by the VCS. Both the methodology and key technologies were presented. For validation, numerical simulations were carried out using 500 simulated 3D–2D correspondence. Simulation results showed that the pose estimation error was $\leq 1\%$ under normal lighting and illuminance, and $\leq 2\%$ under a simulated underground environment. Then, the simulated industrial experiment was established with real underground mining machines. A M16 laser sensor was installed to provide the ground truth. The proposed vision detection method was tested under normal lighting and illuminance, and a simulated underground environment. The result of the experiment was consistent with that of the numerical simulation, which proved the superiority of the approach.

Author Contributions: L.S. and L.Z. conceived and designed the experiments; X.Z. and Y.S. performed the experiments; L.Z. and S.F. developed the algorithm; G.L., Y.S. and L.Z. analyzed the data; L.S. and N.C. wrote the paper; L.Z. translated and re-edited the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science Foundation of China [52004034 and 61903005], and Science and Technology Research Program of Chongqing Municipal Education Commission [KJQN202101413 and KJQN202001404].

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data generated or analyzed during this study are included in the published article, and are available from the corresponding author on reasonable request.

Acknowledgments: This work is supported by the National Natural Science Foundation of China (No. 52004034 and 61903005), and Science and Technology Research Program of Chongqing Municipal Education Commission (No. KJQN202101413 and KJQN202001404). The authors would like to thank the referees for their very useful suggestions and comments.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Oh, X.; Loh, L.; Foong, S.; Bao Andy Koh, Z.; Leong Ng, K.; Kang Tan, P.; Lin Pearlin Toh, P.; Tan, U.-X. Initialisation of Autonomous Aircraft Visual Inspection Systems via CNN-Based Camera Pose Estimation. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May 2021; pp. 11047–11053.
- Li, J.; Lan, F.; Chen, J. Intelligent vehicle visual pose estimation algorithm based on deep learning and parallel computing for dynamic scenes. J. Intell. Fuzzy Syst. 2022, 1–15. [CrossRef]
- Luo, H.; Wang, M.; Wong, P.K.-Y.; Cheng, J.C.P. Full body pose estimation of construction equipment using computer vision and deep learning techniques. *Autom. Constr.* 2020, 110, 103016. [CrossRef]
- 4. Chen, Y.; Tian, Y.; He, M. Monocular human pose estimation: A survey of deep learning-based methods. *Comput. Vis. Image Underst.* 2020, 192, 102897. [CrossRef]
- 5. Du, G.; Wang, K.; Lian, S.; Zhao, K. Vision-based robotic grasping from object localization, object pose estimation to grasp estimation for parallel grippers: A review. *Artif. Intell. Rev.* **2021**, *54*, 1677–1734. [CrossRef]
- 6. Hu, J.; Liu, S.; Liu, J.; Wang, Z.; Huang, H. Pipe pose estimation based on machine vision. *Measurement* 2021, 182, 109585. [CrossRef]
- 7. Zhao, D.; Sun, F.; Wang, Z.; Zhou, Q. A novel accurate positioning method for object pose estimation in robotic manipulation based on vision and tactile sensors. *Int. J. Adv. Manuf. Technol.* **2021**, *116*, 2999–3010. [CrossRef]
- 8. Bala, P.C.; Eisenreich, B.R.; Yoo, S.B.M.; Hayden, B.Y.; Park, H.S.; Zimmermann, J. Automated markerless pose estimation in freely moving macaques with OpenMonkeyStudio. *Nat. Commun.* **2020**, *11*, 4560. [CrossRef]
- 9. Kidziński, Ł.; Yang, B.; Hicks, J.L.; Rajagopal, A.; Delp, S.L.; Schwartz, M.H. Deep neural networks enable quantitative movement analysis using single-camera videos. *Nat. Commun.* 2020, 11, 4054. [CrossRef]
- Valliappan, N.; Dai, N.; Steinberg, E.; He, J.; Rogers, K.; Ramachandran, V.; Xu, P.; Shojaeizadeh, M.; Guo, L.; Kohlhoff, K.; et al. Accelerating eye movement research via accurate and affordable smartphone eye tracking. *Nat. Commun.* 2020, *11*, 1–12. [CrossRef]
- Phisannupawong, T.; Kamsing, P.; Torteeka, P.; Channumsin, S.; Sawangwit, U.; Hematulin, W.; Jarawan, T.; Somjit, T.; Yooyen, S.; Delahaye, D.; et al. Vision-based spacecraft pose estimation via a deep convolutional neural network for noncooperative docking operations. *Aerospace* 2020, 7, 126. [CrossRef]
- 12. Kim, S.; Kim, J.; Park, J.; Lee, D. Vision-Based Pose Estimation of Fixed-Wing Aircraft Using You Only Look Once and Perspectiven-Points. J. Aerosp. Inf. Syst. 2021, 18, 659–664. [CrossRef]
- 13. Horaud, R.; Conio, B.; Leboulleux, O.; Lacolle, B. An analytic solution for the perspective 4-point problem. *Comput. Vis. Graph. Image Process.* **1989**, *47*, 500–507. [CrossRef]
- 14. Moreno-Noguer, F.; Lepetit, V.; Fua, P. EPnP: An Accurate O(*n*) solution to the PnP problem. *Int J Comput Vis.* **2009**, *81*, 155. [CrossRef]

- 15. Li, S.; Xu, C.; Xie, M. A robust O(*n*) solution to the perspective-n-point problem. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 1444–1450. [CrossRef] [PubMed]
- Wang, P.; Xu, G.; Cheng, Y.; Yu, Q. A simple, robust and fast method for the perspective-n-point Problem. *Pattern Recognit. Lett.* 2018, 108, 31–37. [CrossRef]
- 17. Zhou, B.; Chen, Z.; Liu, Q. An efficient solution to the perspective-n-point problem for camera with unknown focal length. *IEEE Access* **2020**, *8*, 162838–162846. [CrossRef]
- 18. Yu, Q.; Xu, G.; Dong, W.; Wang, Z. Solving the perspective-three-point problem with linear combinations: An accurate and efficient method. *Opt. Stuttg.* **2021**, 228, 165740. [CrossRef]
- 19. Zhou, L.; Koppel, D.; Kaess, M. A Complete, Accurate and Efficient Solution for the Perspective-N-Line Problem. *IEEE Robot. Autom. Lett.* **2021**, *6*, 699–706. [CrossRef]
- Liu, L.; Campbell, D.; Li, H.; Zhou, D.; Song, X.; Yang, R. Learning 2D-3D Correspondences to Solve the Blind Perspective-n-Point Problem. Arxiv arXiv:2003.06752.
- Houssein, E.H.; El-din Helmy, B.; Oliva, D.; Elngar, A.A.; Shaban, H. Multi-level Thresholding Image Segmentation Based on Nature-Inspired Optimization Algorithms: A Comprehensive Review. *Metaheuristics Mach. Learn. Theory Appl.* 2021, 239–265.
- 22. Abualigah, L.; Gandomi, A.H.; Elaziz, M.A.; Hussien, A.G.; Khasawneh, A.M.; Alshinwan, M.; Houssein, E.H. Nature-Inspired Optimization Algorithms for Text Document Clustering—A Comprehensive Analysis. *Algorithms* **2020**, *13*, 345. [CrossRef]
- Arockia Dhanraj, J.; Muthiya, S.J.; Subramaniam, M.; Salyan, S.; Chaurasiya, P.K.; Gopalan, A.; Anaimuthu, S. A Comparative Study with J48 and Random Tree Classifier for Predicting the State of Hydraulic Braking System through Vibration Signals. SAE Tech. Pap. 2021, Part 173236, 1–6. [CrossRef]
- 24. Abualigah, L. Group search optimizer: A nature-inspired meta-heuristic optimization algorithm with its results, variants, and applications. *Neural. Comput. Appl.* **2021**, *33*, 2949–2972. [CrossRef]
- Zhao, D.; Liu, L.; Yu, F.; Heidari, A.A.; Wang, M.; Oliva, D.; Muhammad, K.; Chen, H. Ant colony optimization with horizontal and vertical crossover search: Fundamental visions for multi-threshold image segmentation. *Expert Syst. Appl.* 2021, 167, 114122. [CrossRef]
- Zhu, C.; Shen, Y.; Lei, X. Improved Particle Swarm Optimization Approach for Vibration Vision Measurement. Int. J. Pattern Recognit. Artif. Intell. 2021, 35, 2159011. [CrossRef]
- 27. Andrushia, A.D.; Patricia, A.T. Artificial bee colony optimization (ABC) for grape leaves disease detection. *Evol. Syst.* **2020**, *11*, 105–117. [CrossRef]
- 28. Elsisi, M. Improved grey wolf optimizer based on opposition and quasi learning approaches for optimization: Case study autonomous vehicle including vision system. *Artif. Intell. Rev.* **2022**. [CrossRef]
- 29. El-dosuky, M.A.; Shams, M. A Deep Learning Based Cockroach Swarm Optimization Approach for Segmenting Brain MRI Images. In *Medical Informatics and Bioimaging Using Artificial Intelligence*; Springer: Cham, Switzerland, 2022; pp. 3–13.
- Jain, M.; Singh, V.; Rani, A. A novel nature-inspired algorithm for optimization: Squirrel search algorithm. *Swarm Evol. Comput.* 2019, 44, 148–175. [CrossRef]
- 31. Ghasemi-Marzbali, A. A Novel Nature-Inspired Meta-Heuristic Algorithm for Optimization: Bear Smell Search Algorithm; Springer: Berlin/Heidelberg, Germany, 2020; Volume 24, ISBN 0123456789.
- Kumar, N.; Singh, N.; Vidyarthi, D.P. Artificial lizard search optimization (ALSO): A novel nature-inspired meta-heuristic algorithm. Soft Comput. 2021, 25, 6179–6201. [CrossRef]
- Li, M.D.; Zhao, H.; Weng, X.W.; Han, T. A novel nature-inspired algorithm for optimization: Virus colony search. *Adv. Eng. Softw.* 2016, 92, 65–88. [CrossRef]
- 34. Hansen, N.; Müller, S.D.; Koumoutsakos, P. Reducing the Time Complexity of the Derandomized Evolution Strategy with Covariance Matrix Adaptation (CMA-ES). *Evol. Comput.* **2003**, *11*, 1–18. [CrossRef] [PubMed]
- Garrido-Jurado, S.; Muñoz-Salinas, R.; Madrid-Cuevas, F.J.; Marín-Jiménez, M.J. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognit.* 2014, 47, 2280–2292. [CrossRef]
- Saberi-Movahed, F.; Najafzadeh, M.; Mehrpooya, A. Receiving More Accurate Predictions for Longitudinal Dispersion Coefficients in Water Pipelines: Training Group Method of Data Handling Using Extreme Learning Machine Conceptions. *Water Resour. Manag.* 2020, 34, 529–561. [CrossRef]
- 37. Zhang, L.; Xie, X.; Feng, S.; Luo, M. Heuristic dual-tree wavelet thresholding for infrared thermal image denoising of underground visual surveillance system. *Opt. Eng.* **2018**, *57*, 083102. [CrossRef]