

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



4WD Robot Posture Estimation by Radial Multi-View Visual Odometry

Edgar Alonso Martínez-García and
Luz Abril Torres-Méndez

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/intechopen.79130>

Abstract

This chapter presents a four-wheel robot's trajectory tracking model by an extended Kalman filter (EKF) estimator for visual odometry using a divergent trinocular visual sensor. The trinocular sensor is homemade and a specific observer model was developed to measure 3D key-points by combining multi-view cameras. The observer approaches a geometric model and the key-points are used as references for estimating the robot's displacement. The robot's displacement is estimated by triangulation of multiple pairs of environmental 3D key-points. The four-wheel drive (4WD) robot's inverse/direct kinematic control law is combined with the visual observer, the visual odometry model, and the EKF. The robot's control law is used to produce experimental locomotion statistical variances and is used as a prediction model in the EKF. The proposed dead-reckoning approach models the four asynchronous drives and the four damping suspensions. This chapter presents the deductions of models, formulations and their validation, as well as the experimental results on posture state estimation comparing the four-wheel dead-reckoning model, the visual observer, and the EKF with an external global positioning reference.

Keywords: 4WD, visual odometry, trinocular sensor, EKF, visual observer, trajectory estimation

1. Introduction

Autonomous robots obtain precise information about their surroundings by deploying their sensing devices and developing perceptual tasks to accomplish useful missions. Intelligent robots require to concurrently execute multiple functions such as path planning, collision avoidance,

self-localization, tasks scheduling, trajectory control, map building, environment recognition, kinematic/dynamic control, and so forth. Autonomous robots depend on multisensor fusion, which is the process of combining data from the physical sensors into a homogeneous data space.

This chapter presents robot's visual odometry using sensor data obtained from a homemade radial multi-view device (**Figure 1a**). For this case, trinocular sensing is divergent; hence, an inherent problem refers to different perspectives in each camera. Besides, the partial overlap between adjacent cameras allows sharing approximately 25% of the total sensing angles, which is too reduced and limits extracting numerous relevant features for data fusion affecting to infer consistent information. Besides perspective, radial cameras yield differences of scale, skew, rotation, and lighting intensities. To cope with this problem, this chapter deduces a geometric trinocular sensor model to directly measure 3D data by combining divergent pairs, the central camera with one of the two lateral cameras (**Figure 1b**). The robot's state vector (posture) is recursively estimated by a visual odometry model that triangulates multiple pairs of key-points. Thus, an EKF uses the 3D odometry model and estimates the robot's position. The mobile robot is a four-wheel drive (4WD) modeled by a differential control law involving the four passive damping suspensions to infer accurate positions.

Parallel trinocular stereo systems had been deployed either to detect the ground [1], or to estimate motion [2]. There are reported works on motion estimation with binocular divergent systems [3], trinocular divergence for visual odometry [4], and divergent visual simultaneous localization and mapping (SLAM) [5]. As a difference from the active sensing modalities for localization [6], and concurrent localization and mapping with parallel multi-view [7], this chapter intends to estimate the posture of a rolling vehicle by exploiting feedback of the rich data fusion that a divergent trinocular sensor provides. Numerous visual odometry algorithms had been reported, using stereo cameras [8], matching multi-frame features [9] and 3D point cloud [10]. Some outdoor visual odometry approaches for urban [11] environments estimate motion tracking by extraction of visual feature points. There are numerous works combining the benefits of visual SLAM algorithms [12–14] with visual odometry [15], detecting geometrical features [16].

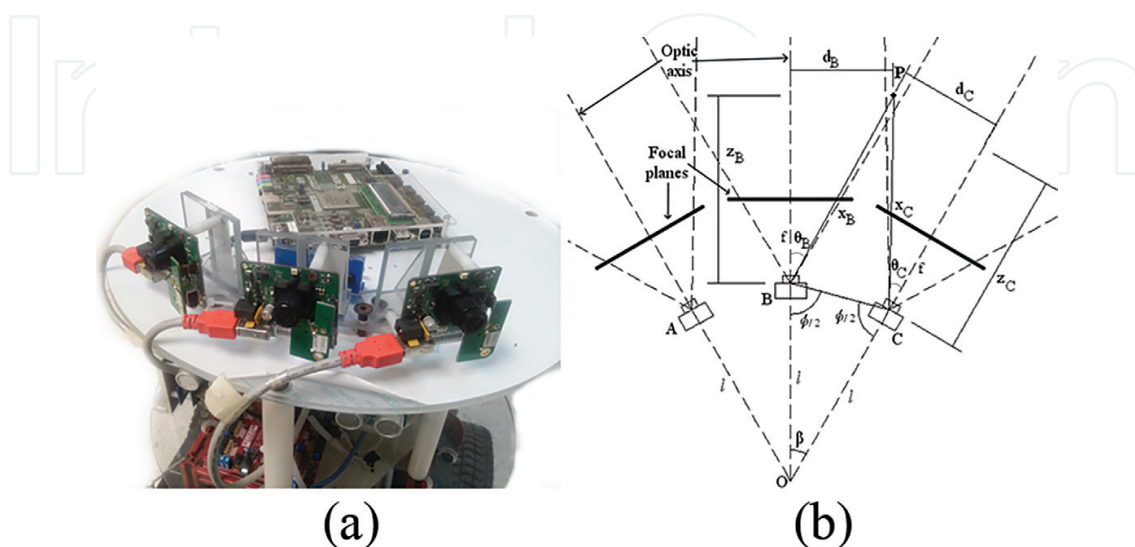


Figure 1. Robot's trinocular sensor. (a) Trinocular sensor onboard. (b) Camera's geometric divergence (top view).

This chapter is organized into to the following sections. Section 2 deduces the sensor fusion observer modeling the trinocular system geometry. Section 3 models the 4WD direct/inverse kinematic solutions. Section 4 deduces the visual odometry formulation and EKF-based control state and estimation. Finally, conclusions are provided in Section 5.

2. Trinocular sensing model

This section describes the divergent multi-view geometric model, which basically combines the data of a pair of cameras radially arranged. In addition, this section presents an algebraic analysis of the lateral cameras' alignment and correction w.r.t. the central camera. The fundamental geometrical relationship of the system divergence was experimentally studied by deploying a homemade prototype onboard a mobile robot, see **Figure 2a**. Cameras with homogeneous intrinsic parameters are assumed, and cameras are mechanically fixed epipolar. The sensor model's purpose is to determine the depth information of a point in the scene $\mathbf{p} = (x, y, z)^\top$, which is projected onto the overlapping area of a divergent pair. The proposed multi-view geometric model combines data using the central camera as the reference (**Figure 1b**). The focal plane in cameras A, C are perspective transformed, in order to align them epipolar and coplanar w.r.t. the central reference B . As seen in **Figure 1b**, a point \mathbf{p} is projected over two focal planes, for instance, at column x_C of lateral camera C , and at x_B of camera B . Thus, an isosceles triangle PBC is formed. For the triangle OBC , let β be the angle between the cameras' centers B and C , as deployed by expression (1). Let $\phi/2$ be the remaining angle of OBC , where the inner angles' total sum is π . By expressing $\beta + \phi = \pi$ and dropping off $\phi = \pi - \beta$, we easily deduce that $\frac{\phi}{2} = \frac{\pi}{2} - \frac{\beta}{2}$. The geometrical distance \overline{BC} is calculated by triangulation using the law of sines, with known distance l that converges to O . The linear distance between adjacent sensors BC commonly oriented w.r.t. the center O is

$$\overline{BC} = \frac{l \sin \beta}{\sin \left(\frac{\pi}{2} - \frac{\beta}{2} \right)}. \quad (1)$$

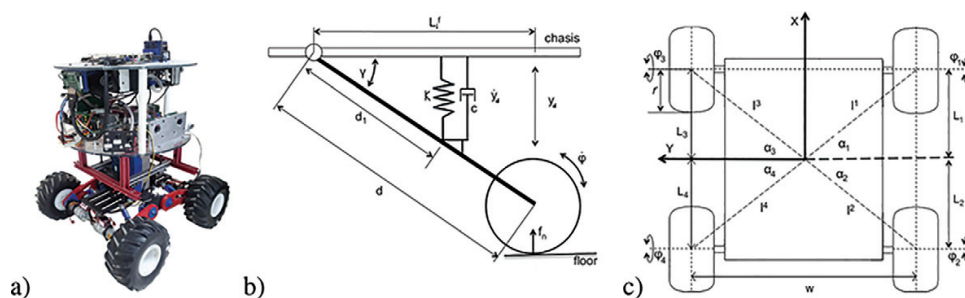


Figure 2. 4WD kinematics. (a) Deployed robot. (b) Damping system. (c) Robot's view from below.

To calculate the Cartesian coordinates of \mathbf{p} , let us state that the point \mathbf{p} is projected through the horizontal coordinate x_B , and on camera B angle's θ_B , and focal distance f_B as expressed by

$$\theta_B = \tan^{-1}\left(\frac{x_B}{f_B}\right) \quad \text{and} \quad \theta_C = \tan^{-1}\left(\frac{x_C}{f_C}\right). \quad (2)$$

The complementary angles B and C are modeled by

$$\angle B = \frac{\pi}{2} - \theta_B + \frac{\beta}{2} \quad \text{and} \quad \angle C = \frac{\pi}{2} - \theta_C + \frac{\beta}{2}. \quad (3)$$

In the triangle BCO , the angle at point \mathbf{p} is obtained by equivalence of similar triangles $\angle P = \theta_B + \theta_C - \beta$. Thus, to estimate the range of the radial system B and C w.r.t. \mathbf{p} , the linear distance is calculated by the law of sines:

$$\frac{\overline{BC}}{\sin \angle P} = \frac{\overline{CP}}{\sin \angle B} \quad \text{and} \quad \overline{CP} = \frac{\overline{BC} \sin \angle B}{\sin \angle P}. \quad (4)$$

Thus, for the other cameras' pair, similar expressions are stated

$$\frac{\overline{BC}}{\sin \angle P} = \frac{\overline{BP}}{\sin \angle C} \quad \text{and} \quad \overline{BP} = \frac{\overline{BC} \sin \angle C}{\sin \angle P}. \quad (5)$$

Hence, the model to express depth information is given by $z_B = \overline{BP} \cos \theta_B$. By substituting \overline{BP} and θ_B , the model is further specified by

$$\Lambda_1 = \frac{l \sin \beta}{\sin\left(\frac{\pi - \beta}{2}\right) \sin(\theta_B - \theta_C - \beta)},$$

where,

$$z_B = \Lambda_1 \sin\left(\frac{\pi + \beta}{2} - \theta_C\right) \cos\left(\tan^{-1}\left(\frac{x_B}{f_B}\right)\right). \quad (6)$$

In addition, the range between camera C and \mathbf{p} is defined by $z_C = \overline{CP} \cos \theta_C$. Thus, substituting \overline{CP} and θ_C , we define

$$z_C = \Lambda_1 \sin\left(\frac{\pi + \beta}{2} - \theta_B\right) \cos\left(\tan^{-1}\left(\frac{x_C}{f_C}\right)\right). \quad (7)$$

Using the depth models z_B and z_C , the distances d_B and d_C w.r.t. \mathbf{p} are estimated, such that $d_B = z_B \tan(\theta_B)$. Hence,

$$d_B = z_B \tan\left(\tan^{-1}\left(\frac{x_B}{f_B}\right)\right) \quad \text{or} \quad d_B = \frac{z_B x_B}{f_B} \quad (8)$$

and being $d_C = z_C \tan(\theta_C)$, by substituting θ_C from expression (2) we have

$$d_C = z_C \tan \left(\tan^{-1} \left(\frac{x_C}{f_C} \right) \right) \quad \text{or} \quad d_C = \frac{z_C x_C}{f_C}. \quad (9)$$

Furthermore, the algebraic deduction along the Y component for the equalities $h_B f_B = z_B y_B$ and $h_C f_C = z_C y_C$, w.r.t \mathbf{p} using distances h_B and h_C , is obtained by

$$h_B = \frac{z_B y_B}{f_B} \quad \text{and} \quad h_C = \frac{z_C y_C}{f_C},$$

thus the following term is stated as

$$\Psi = \frac{1}{f_B} \left(\frac{l \sin \beta}{\sin \left(\frac{\pi}{2} - \frac{\beta}{2} \right)} \right) \cos \left(\tan^{-1} \left(\frac{x_B}{f_B} \right) \right).$$

Therefore, the geometry vector model for camera B w.r.t. camera C , with substitution of the terms Ψ , z_B , and z_C in robot's inertial frame R , produce the next expression:

$$\mathbf{p}_{BC}^R = \frac{\Psi \sin \left(\frac{\pi + \beta}{2} - \theta_C \right)}{\sin(\theta_B + \theta_C - \beta)} \begin{pmatrix} x_B \\ y_B \\ f_B \end{pmatrix} \quad (10)$$

and the same model is enhanced for camera B , using the geometry of cameras A and B by

$$\mathbf{p}_{AB}^R = \frac{\Psi \sin \left(\frac{\pi + \beta}{2} - \theta_A \right)}{\sin(\theta_A + \theta_B - \beta)} \begin{pmatrix} x_B \\ y_B \\ f_B \end{pmatrix}. \quad (11)$$

Hence, the arbitrary point $\mathbf{p}_{ABC} \in \mathbb{R}^3$ is projected onto cameras AB , or onto cameras BC . In order to express a general formula, let us define the following theorem.

Theorem 1 (Trinocular depth model). *Let camera B be the reference for either divergent camera A or C . A point coordinates w.r.t. camera A is $\mathbf{p}_{AB} = (x_A, y_A, z)^T$, and w.r.t. camera C is $\mathbf{p}_{AC} = (x_C, y_C, z)^T$. Hence, the general depth coordinate model for x and y for any divergent pair is*

$$x, y_{A,C} = \frac{\Psi x_B \sin \left(\frac{\pi + \beta}{2} - \theta_{A,C} \right)}{\sin(\theta_{A,C} + \theta_B - \beta)}, \quad (12)$$

for coordinate z ,

$$z_{A,C} = \frac{\Psi f_B^2 \sin \left(\frac{\pi + \beta}{2} - \theta_{A,C} \right)}{\sin(\theta_{A,C} + \theta_B - \beta)}. \quad (13)$$

The four points shown by the three cameras may illustrate their transformation, experimentally developed at 1-m distance between the robot and the marks.

3. 4WD dead-reckoning controller

Since the visual trinocular approach uses an exteroceptive sensor, we decided to challenge its detection and tracking capabilities with a robot having high holonomic properties. A 4WD robot's locomotion is prone to experience frequent swift turns resulting in numerous slippages. Thus, a 4WD has to depend more on exteroceptive rather than inner measurements. Comparatively, inner 4WD odometry differs greatly from external visual measurement to infer posture. The proposed dead-reckoning system obtains speed measurements by deploying odometer readings of the four asynchronous drives (**Figure 2**). A 4WD system is considerably different from a conventional differential dual approach. Moreover, four passive mass-spring-damper suspensions are included in this system (**Figure 2b**), which varies the inter-wheel distances over time. Particularly, the robot's 4WD and passive suspensions make the posture observations challenging.

The robot's dead-reckoning model is fundamental to sense and control position used as feedback, providing motion description as a kinematic reference to match the visual observations when estimating the robot's motion. The positioning and trajectory control [17], as well as the type of kinematic analysis [18] and the dynamic suspension [19] in this type of robot have been previously reported. The robot's instantaneous speed v_t (m/s) and yaw rate ω_t (rad/s) depend on the four wheels' asynchronous rolling motion, $\dot{\varphi}_1, \dot{\varphi}_2, \dot{\varphi}_3, \dot{\varphi}_4$. For a wheel's encoder, the velocity model approaches measurements by high-precision numerical derivatives (central divided differences) of the rotary angle φ_t (rad) w.r.t. time t , such that

$$\dot{\varphi}_t = \frac{\pi}{6Rt} [\eta_{t+2} + 7\eta_{t+1} + 7\eta_{t-1} - \eta_{t-2}], \quad (14)$$

where the wheel's angular speed $\dot{\varphi}$ is measured through pulse detection η (dimensionless) of encoder's resolution R (pulses/rev); thus, the robot's instantaneous velocity is modeled by the averaged wheel speed, with wheels of nominal radius r (m)

$$v_t = \frac{r}{4} \sum_{i=1}^4 \dot{\varphi}_i. \quad (15)$$

Further, the differential velocity \hat{v}_t expresses the lateral speeds' difference that yields ω_t . Thus, \hat{v}_t is formulated by the expression

$$\hat{v}_t = r(\dot{\varphi}_1 + \dot{\varphi}_2 - \dot{\varphi}_3 - \dot{\varphi}_4). \quad (16)$$

This model describes that the rotary motion of $\dot{\varphi}_1$ and $\dot{\varphi}_2$ contributes to robot's $+\omega_t$ (counter-clockwise sense). Likewise, $\dot{\varphi}_3$ and $\dot{\varphi}_4$ contribute to robot's $-\omega_t$ (clockwise sense). Therefore, ω_t is yielded by the lateral speed component $\hat{v}_t \cos(\alpha_i)$ (see **Figure 2b**) modeled by

$$\omega_t = \frac{\hat{v} \cos(\alpha_i)}{l_i}. \quad (17)$$

The previous equation expresses the conservation of angular motion, and the wheel's contact point turns w.r.t. the robot's center,

$$\cos(\alpha_i) = \frac{W}{2l_i}, \quad (18)$$

where for each length l_i there is an asynchronous model,

$$l_1 = \frac{\sqrt{W^2 + (L_1 + L_4)^2}}{2}, \quad l_2 = \frac{\sqrt{W^2 + (L_2 + L_3)^2}}{2},$$

as well as

$$l_3 = \frac{\sqrt{W^2 + (L_3 + L_2)^2}}{2}, \quad l_4 = \frac{\sqrt{W^2 + (L_4 + L_1)^2}}{2}.$$

Thus, substituting $\cos(\alpha_i)$ and l_i into ω_t and by considering both clockwise and counterclockwise motions, the robot's angular speed is

$$\omega_t = \sum_{i=1}^2 \frac{rW\dot{\phi}_i}{l_i^2} - \sum_{i=3}^4 \frac{rW\dot{\phi}_i}{l_i^2}. \quad (19)$$

The longitudinal contact point's distance l_i (m) takes as reference the robot's geometric center. When l_i varies, the contact point's position L_i changes.

$$L_i = d \cos(\arcsin(\gamma_i)), \quad (20)$$

where γ_i represents the vertical motion along the suspension,

$$\gamma_i = \arcsin\left(\frac{\Delta y}{d_1}\right). \quad (21)$$

From **Figure 2a**, the vertical motion y_d is modeled assuming critical damping motion for a general spring-mass-damper system. The suspension is modeled by the following second-order homogeneous differential equation:

$$m\ddot{y}_d + \kappa_2\dot{y}_d + \kappa_1y_d = 0, \quad (22)$$

where the elastic spring restitution coefficient is κ_1 (kg/s²). The damping coefficient is κ_2 (kg/s). The restitution force $m\ddot{y}_d$ counteracts the vertical oscillatory damping effects. The oscillatory velocity and acceleration are denoted by \dot{y}_d and \ddot{y}_d , respectively. Thus, solving the second-order differential equation as a first-order equation such that $\kappa_2\dot{y}_d = -\kappa_1y_d$,

$$\int_{y_d} \frac{dy_d}{y_d} = -\frac{\kappa_1}{\kappa_2} \int_t dt, \quad (23)$$

hence

$$\ln(y_d) = -\frac{\kappa_1}{\kappa_2} t + c, \quad \zeta = -\frac{\kappa_1}{\kappa_2}, \quad (24)$$

with integration constant $c = 0$ for analysis purpose. The suspension's elongation derivatives as functions of time are

$$y_d = e^{\zeta t}, \quad \dot{y}_d = \zeta e^{\zeta t}, \quad \ddot{y}_d = \zeta^2 e^{\zeta t}. \quad (25)$$

Substituting the previous expression in (22),

$$m\zeta^2 e^{\zeta t} + \kappa_2 \zeta e^{\zeta t} + \kappa_1 e^{\zeta t} = 0, \quad (26)$$

and by algebraically simplifying, the characteristic equation is

$$\zeta^2 + \frac{\kappa_2}{m} \zeta + \frac{\kappa_1}{m} = 0, \quad (27)$$

and its analytic solution is

$$\zeta_{1,2} = \frac{-\kappa_2}{2m} \pm \frac{\sqrt{\left(\frac{\kappa_2}{m}\right)^2 - 4\frac{\kappa_1}{m}}}{2}. \quad (28)$$

As we assume a critically damped system, $\left(\frac{\kappa_2}{m}\right)^2 = 4\left(\frac{\kappa_1}{m}\right)$ and there is only one real root solution, such that

$$\zeta = -\frac{\kappa_2}{2m}. \quad (29)$$

Therefore, the damping motion is analytically solved by

$$y_d(t) = Ae^{\zeta t}, \quad (30)$$

where A (m) is the elongation amplitude (m) parameter for the suspension system. Moreover, in this type of robotic platform, the four asynchronous drives simultaneously produce slip/skid motions that are advantageously used to maneuver the robot. This approach proposes inferring the instantaneous Z-turn axis location $(x_R, y_R)^T$. The Z-turn axis is movable in the square region bounded by the wheels' contact point. The Z-turn axis is expressed by the first-order derivatives

$$\dot{x}_R = \frac{rW}{4v_{max}} (\ddot{\varphi}_1 - \ddot{\varphi}_2 - \ddot{\varphi}_3 + \ddot{\varphi}_4) \quad (31)$$

and

$$\dot{y}_R = \frac{rL}{4v_{max}} (-\ddot{\varphi}_1 - \ddot{\varphi}_2 + \ddot{\varphi}_3 + \ddot{\varphi}_4). \quad (32)$$

There is a maximal allowable Z-turn displacement speed v_{max} . Hereafter, with four independent equations, the control positioning system is instantaneously computed. The robot control vector is $\dot{\mathbf{u}}_R = (\dot{v}_t, \dot{\omega}_t, \dot{x}_R, \dot{y}_R)^\top$, and the control transition matrix Λ_t has the elements $\lambda_1 = r/4$, $\lambda_{2i} = rW/l_i^2$, $\lambda_3 = rW/(4v_{max})$, and $\lambda_4 = rL/(4v_{max})$. Thus, the forward kinematics solution is $\dot{\mathbf{u}}_R = \Lambda_t \cdot \dot{\mathbf{\Omega}}$ or

$$\begin{pmatrix} \dot{v}_t \\ \dot{\omega}_t \\ \dot{x}_R \\ \dot{y}_R \end{pmatrix} = \begin{pmatrix} \lambda_1 & \lambda_1 & \lambda_1 & \lambda_1 \\ \lambda_{21} & \lambda_{22} & -\lambda_{23} & -\lambda_{24} \\ \lambda_3 & -\lambda_3 & -\lambda_3 & \lambda_3 \\ -\lambda_4 & -\lambda_4 & \lambda_4 & \lambda_4 \end{pmatrix} \cdot \begin{pmatrix} \ddot{\varphi}_1 \\ \ddot{\varphi}_2 \\ \ddot{\varphi}_3 \\ \ddot{\varphi}_4 \end{pmatrix}. \quad (33)$$

In addition, to inversely solve this matrix system, the analytical solution represents the vector of independent control rotary variables $\dot{\mathbf{\Omega}}_t = (\ddot{\varphi}_1, \ddot{\varphi}_2, \ddot{\varphi}_3, \ddot{\varphi}_4)^\top$. Thus, let us define $\lambda_w = \lambda_1\lambda_3\lambda_4$, $\lambda_A = \lambda_{23} - \lambda_{24}$, $\lambda_B = \lambda_{22} - \lambda_{21}$, $\lambda_C = \lambda_{21} - \lambda_{23}$, $\lambda_D = \lambda_{24} - \lambda_{22}$, $\lambda_E = \lambda_{21} + \lambda_{24}$, and $\lambda_F = \lambda_{22} - \lambda_{23}$.

$$\ddot{\varphi}_1 = \frac{\lambda_3\lambda_4\lambda_D\dot{v} + 2\lambda_w\dot{\omega} - \lambda_1\lambda_4\lambda_A\dot{x}_R + \lambda_1\lambda_3\lambda_F\dot{y}_R}{2\lambda_w(\lambda_{21} - \lambda_{22} - \lambda_{23} + \lambda_{24})}, \quad (34)$$

$$\ddot{\varphi}_2 = \frac{\lambda_3\lambda_4\lambda_C\dot{v} - 2\lambda_w\dot{\omega} + \lambda_1\lambda_4\lambda_A\dot{x}_R - \lambda_1\lambda_3\lambda_E\dot{y}_R}{2\lambda_w(\lambda_{21} - \lambda_{22} - \lambda_{23} + \lambda_{24})}, \quad (35)$$

$$\ddot{\varphi}_3 = \frac{\lambda_3\lambda_4\lambda_D\dot{v} + 2\lambda_w\dot{\omega} + \lambda_1\lambda_4\lambda_B\dot{x}_R + \lambda_1\lambda_3\lambda_E\dot{y}_R}{2\lambda_w(\lambda_{21} - \lambda_{22} - \lambda_{23} + \lambda_{24})} \quad (36)$$

and

$$\ddot{\varphi}_4 = \frac{\lambda_3\lambda_4\lambda_C\dot{v} - 2\lambda_w\dot{\omega} - \lambda_1\lambda_4\lambda_B\dot{x}_R - \lambda_1\lambda_3\lambda_F\dot{y}_R}{2\lambda_w(\lambda_{21} - \lambda_{22} - \lambda_{23} + \lambda_{24})}. \quad (37)$$

The matrix form of the inverse analytical solution for all wheels' speed under damping variations is stated as $\dot{\mathbf{\Omega}} = \Lambda_t^{-1} \cdot \dot{\mathbf{u}}_t$ or

$$\dot{\Omega}_t = \begin{pmatrix} \frac{\lambda_3 \lambda_4 \lambda_D}{2\lambda_w \lambda_G} & \frac{2\lambda_w}{2\lambda_w \lambda_G} & \frac{-\lambda_1 \lambda_4 \lambda_A}{2\lambda_w \lambda_G} & \frac{\lambda_1 \lambda_3 \lambda_F}{2\lambda_w \lambda_G} \\ \frac{\lambda_3 \lambda_4 \lambda_C}{2\lambda_w \lambda_G} & \frac{-2\lambda_w}{2\lambda_w \lambda_G} & \frac{\lambda_1 \lambda_4 \lambda_A}{2\lambda_w \lambda_G} & \frac{-\lambda_1 \lambda_3 \lambda_E}{2\lambda_w \lambda_G} \\ \frac{\lambda_3 \lambda_4 \lambda_D}{2\lambda_w \lambda_G} & \frac{2\lambda_w}{2\lambda_w \lambda_G} & \frac{\lambda_1 \lambda_4 \lambda_B}{2\lambda_w \lambda_G} & \frac{\lambda_1 \lambda_3 \lambda_E}{2\lambda_w \lambda_G} \\ \frac{\lambda_3 \lambda_4 \lambda_C}{2\lambda_w \lambda_G} & \frac{-2\lambda_w}{2\lambda_w \lambda_G} & \frac{-\lambda_1 \lambda_4 \lambda_B}{2\lambda_w \lambda_G} & \frac{-\lambda_1 \lambda_3 \lambda_F}{2\lambda_w \lambda_G} \end{pmatrix} \cdot \begin{pmatrix} \dot{v}_t \\ \dot{\omega}_t \\ \dot{x}_R \\ \dot{y}_R \end{pmatrix}, \quad (38)$$

where $\lambda_G = \lambda_{21} - \lambda_{22} - \lambda_{23} + \lambda_{24}$.

4. State estimation and feedback position control

This section formulates a deterministic geometric model for visual odometry and the state estimation by an EKF. The proposed model combines pairs of key-points at times t and $t - 1$. The robot's displacements are deduced by inverse geometric triangulations to feed forward an EKF and estimate the robot's posture.

In **Figure 3a**, the instantaneous angle α_{t-1} is formed by a pair of key-points defined by

$$\alpha_{t-1} = |\theta_{t-1}^a| + |\theta_{t-1}^b| \quad (39)$$

and such key-points' distance c_{t-1} is defined by

$$c_{t-1} = \sqrt{(\delta_{t-1}^a)^2 + (\delta_{t-1}^b)^2 - 2\delta_{t-1}^a \delta_{t-1}^b \cos \alpha_{t-1}}. \quad (40)$$

The angle $\beta_{t-1}^{a,b}$ of either key-point a or b is calculated by the law of sines,

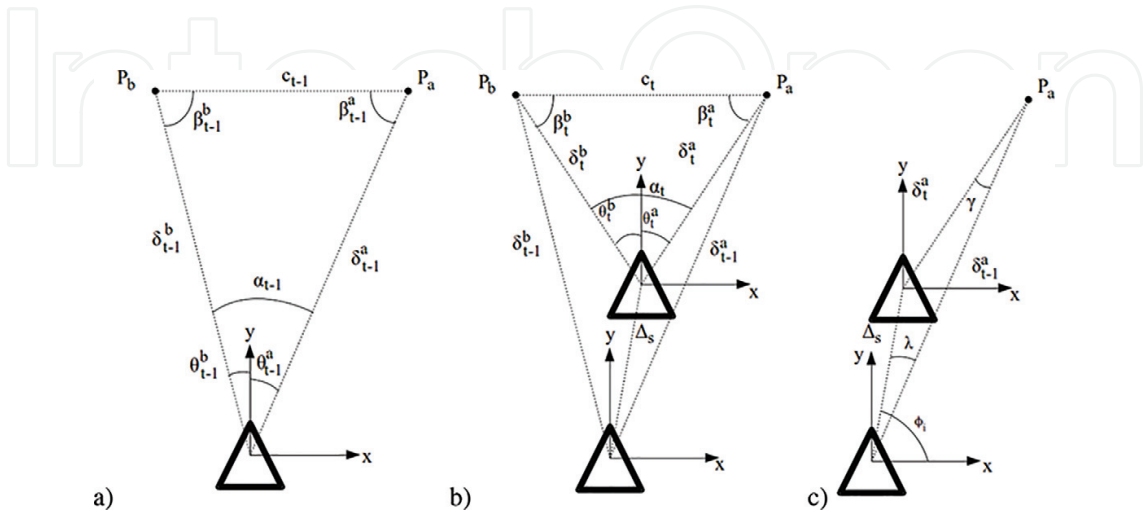


Figure 3. Robot's visual odometry. (a) Robot's key-point pair observed at $t - 1$. (b) Same pair observed at t . (c) Robot's displacement Δ_s by triangulation of key-points p_a .

$$\beta_{t-1}^{a,b} = \arcsin\left(\frac{\delta_{t-1}^{b,a} \sin \alpha_{t-1}}{c_{t-1}}\right). \quad (41)$$

However, at time t in **Figure 3b**, the instantaneous angle α_t is obtained by

$$\alpha_t = |\theta_t^a| + |\theta_t^b|, \quad (42)$$

with the key-point's distance c_t as

$$c_t = \sqrt{(\delta_t^a)^2 + (\delta_t^b)^2 - 2\delta_t^a \delta_t^b \cos(\alpha_t)}, \quad (43)$$

which is used to obtain the angle $\beta_t^{a,b}$ of the key-point a or b at actual time

$$\beta_t^{a,b} = \arcsin\left(\frac{\delta_t^{b,a} \sin \alpha_t}{c_t}\right). \quad (44)$$

Further, the differential angle $\hat{\beta}$ is defined by triangulation of previous and actual poses and an arbitrary 3D point $\mathbf{p}_{a,b}$ (**Figure 3c**),

$$\hat{\beta} = \beta_{t-1}^{a,b} - \beta_t^{a,b}. \quad (45)$$

Proposition 1 (Triangulation odometric displacement). The robot's displacement Δ_s (**Figure 3c**) that is inferred by triangulation of visual key-points over time is

$$\Delta_s = \sqrt{\delta_{t-1}^a + \delta_t^a - 2\delta_{t-1}^a \delta_t^a \cos(\hat{\beta})}. \quad (46)$$

The triangulation angle λ is calculated by the law of sines,

$$\lambda^{a,b} = \arcsin\left(\frac{\delta_t^{a,b} \sin(\hat{\beta})}{\Delta_s^{a,b}}\right) \quad (47)$$

and the orientation angle for each reference a is

$$\phi^{a,b} = \lambda^{a,b} + \left(\frac{\pi}{2} - \theta_{t-1}^{a,b}\right), \quad (48)$$

which is required to know the X displacement

$$\Delta_x = \Delta_s^{a,b} \cos(\phi^{a,b}), \quad (49)$$

as well as the Y displacement

$$\Delta_y = \Delta_s^a \sin \phi^a. \quad (50)$$

When obtaining numerous key-point pairs simultaneously, the total robot's displacement is an averaged value of the displacements yielded by all key-point pairs,

$$\Delta_{x_t} = \frac{\sum_{i=1}^n \Delta x_i}{n} \quad \text{and} \quad \Delta_{y_t} = \frac{\sum_{i=1}^n \Delta y_i}{n}. \quad (51)$$

Therefore, without loss of generality, for state estimation, let us assume a nonlinear robot's model state vector

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{u}_k, \mathbf{w}_k), \quad (52)$$

where the state vector is $\mathbf{x} = (x, y, \theta, v, \omega)$, and combined with a nonstationary state transition matrix \mathbf{A}_t , such that $\mathbf{x}_k = \mathbf{A}_t \cdot \mathbf{x}_{k-1}$ or

$$\mathbf{x}_k = \begin{pmatrix} 1 & 0 & 0 & \cos(\theta)\Delta t & 0 \\ 0 & 1 & 0 & \sin(\theta)\Delta t & 0 \\ 0 & 0 & 1 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ \theta \\ v \\ \omega \end{pmatrix}, \quad (53)$$

by developing the dot product from previous expression, we obtain

$$\mathbf{x}_k = \begin{pmatrix} x_{k-1} + v \cos(\theta)\Delta t \\ x_{k-1} + v \sin(\theta)\Delta t \\ \theta_{k-1} + \omega\Delta t \\ v \\ \omega \end{pmatrix}. \quad (54)$$

The measurement model requires the displacements that were inferred through key-point triangulation

$$\mathbf{z}_k = h(\mathbf{x}_k, \mathbf{v}_k), \quad (55)$$

where \mathbf{w}_k and \mathbf{v}_k are the process and measurement noise models, respectively. These are statistically independent and supposed to have a Gauss distribution with zero average value and known variance. To approximate the nonlinear robot's measurement model, a linearized first-order approximation by the expansion of the Taylor series is used, and a linear approximation of a function is built, with slope obtained through partial derivatives by

$$f'(u_t, x_{t-1}) = \frac{\partial f(u_t, x_{t-1})}{\partial x_{t-1}}. \quad (56)$$

Thus, the linearized models of the process and measurement are defined next in (54) and (55), such that

$$f(\mathbf{x}) = f(\hat{\mathbf{x}}) + \underbrace{f'(\hat{\mathbf{x}})}_{=J} (\mathbf{x} - \hat{\mathbf{x}}) \quad (57)$$

and

$$h(\mathbf{x}) = h(\hat{\mathbf{x}}) + \underbrace{h'(\hat{\mathbf{x}})}_{=H} (\mathbf{x} - \hat{\mathbf{x}}). \quad (58)$$

In addition, the EKF's prediction models (56) and the correction models (57) are formulated and linearized as

$$\hat{\mathbf{x}}_k^- = f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, 0) \quad (59)$$

and

$$\mathbf{P}_k^- = \mathbf{A}_k \mathbf{P}_{k-1} \mathbf{A}_k^\top + \mathbf{W}_k \mathbf{Q}_{k-1} \mathbf{W}_k^\top. \quad (60)$$

Moreover, the recursive Kalman gain for system convergence is

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^\top (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^\top + \mathbf{V}_k \mathbf{R}_k \mathbf{V}_k^\top)^{-1} \quad (61)$$

and the state vector of the system is described by

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{z}_k - \mathbf{H} \hat{\mathbf{x}}_k^-), \quad (62)$$

with covariance matrix of the system

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^-. \quad (63)$$

Thus, hereafter, the vector and matrix models describing the proposed robot's system are formulated and incorporated into the conventional EKF. Let us define the robot's pose vector $\mathbf{x} = (x_t, y_t, \theta_t)^\top$. The control vector is comprised of the robot's absolute and angular speeds, $\mathbf{u}_k = (v, \omega)^\top$. Furthermore, the observation vector with sensor measurement $\mathbf{z}_k = (X_p, Y_p, Z_p)^\top$. Eventually, the process noise vector $\mathbf{w}_k = (w_x, w_y, w_\theta)^\top$. The measurement noise vector $\mathbf{v}_k = (v_{X_p}, v_{Y_p}, v_{Z_p})^\top$.

Therefore, from the displacement equation (46), which arises from exteroceptive observations, the robot's Cartesian displacements are

$$\Delta x = (\Delta s) \cos \left(\theta_{k-1} + \lambda + \left(\frac{\pi}{2} - \theta_{t-1}^a \right) \right) \quad (64)$$

and

$$\Delta y = (\Delta s) \sin \left(\theta_{k-1} + \lambda + \left(\frac{\pi}{2} - \theta_{t-1}^a \right) \right), \quad (65)$$

as well as $\Delta \theta$ is given by

$$\Delta \theta = \lambda + \left(\frac{\pi}{2} - \theta_{t-1}^a \right). \quad (66)$$

By substituting an averaged Cartesian displacement, one considers n key-point observations to calculate the recursive noisy state vector \mathbf{x}_k . The Jacobian matrix \mathbf{J}_k of the robot's temporal matrix state transition $\mathbf{A}_k \cdot \mathbf{x}_k$, where $x_k = x_{k-1} + v \cos(\theta) \Delta t$, $y_k = y_{k-1} + v \sin(\theta) \Delta t$, and $\theta_k = \theta_{k-1} + \omega \Delta t$ is stated by

$$\mathbf{J} = \frac{\partial \mathbf{A}_k \cdot \mathbf{x}_k}{\partial \mathbf{x}_{k-1}} = \begin{pmatrix} \frac{\partial x}{\partial x} & \frac{\partial x}{\partial y} & \frac{\partial x}{\partial \theta} & \frac{\partial x}{\partial v} & \frac{\partial x}{\partial \omega} \\ \frac{\partial y}{\partial x} & \frac{\partial y}{\partial y} & \frac{\partial y}{\partial \theta} & \frac{\partial y}{\partial v} & \frac{\partial y}{\partial \omega} \\ \frac{\partial \theta}{\partial x} & \frac{\partial \theta}{\partial y} & \frac{\partial \theta}{\partial \theta} & \frac{\partial \theta}{\partial v} & \frac{\partial \theta}{\partial \omega} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} & \frac{\partial v}{\partial \theta} & \frac{\partial v}{\partial v} & \frac{\partial v}{\partial \omega} \\ \frac{\partial \omega}{\partial x} & \frac{\partial \omega}{\partial y} & \frac{\partial \omega}{\partial \theta} & \frac{\partial \omega}{\partial v} & \frac{\partial \omega}{\partial \omega} \end{pmatrix} = \begin{pmatrix} 1 & 0 & -v \sin(\theta) \Delta t & \cos(\theta) \Delta t & 0 \\ 0 & 1 & v \cos(\theta) \Delta t & \sin(\theta) \Delta t & 0 \\ 0 & 0 & 1 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \quad (67)$$

Thus, a measurement is a 3D point arising from either divergent pair Eq. (10) or (11) and deployed by Proposition 1.1. Thus, the robot's measurement vector model \mathbf{z} includes noise measurements. The Jacobian matrix \mathbf{H} of the expected state model w.r.t. measurements is defined by

$$\mathbf{H} = \frac{\partial \mathbf{h}}{\partial \mathbf{x}} = \begin{pmatrix} \frac{\partial \Delta x}{\partial x_{k-1}} & \frac{\partial \Delta x}{\partial y_{k-1}} & \frac{\partial \Delta x}{\partial \theta_{k-1}} \\ \frac{\partial \Delta y}{\partial x_{k-1}} & \frac{\partial \Delta y}{\partial y_{k-1}} & \frac{\partial \Delta y}{\partial \theta_{k-1}} \\ \frac{\partial \Delta \theta}{\partial x_{k-1}} & \frac{\partial \Delta \theta}{\partial y_{k-1}} & \frac{\partial \Delta \theta}{\partial \theta_{k-1}} \end{pmatrix}. \quad (68)$$

The process noise covariance matrix \mathbf{Q}_k is defined by

$$\mathbf{Q}_k | 2\pi \Sigma |^{-1/2} \exp \left\{ 0.5 (\mathbf{z} - \boldsymbol{\mu})^\top \Sigma^{-1} (\mathbf{z} - \boldsymbol{\mu}) \right\}. \quad (69)$$

Let us define the nonstationary covariance matrix \mathbf{P} ,

$$\mathbf{P} = \begin{pmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_\theta^2 \end{pmatrix}, \quad (70)$$

the matrix diagonal variances are experimental measurements that describe the trend of the robot motion's error.

The robot's motion covariance matrix was obtained experimentally through 500 tests—straight motion, right turns, left turns, clockwise and counterclockwise rotations, with $N = 100$ tests for each type of motion. Exteroceptive sensing devices onboard are tied to the robot's geometry of motion, and with their observations the robot's posture can be estimated and therefore matched with the robot's deterministic kinematic model. From Section 3, the inverse (38) and direct (33) models were used experimentally to obtain the following statistical covariance about the measurement model

$$\sigma_x^2 = \frac{1}{N} \sum_{i=1}^N \left(\int_t (\lambda_1, \lambda_1, \lambda_1, \lambda_1)^\top \cdot \dot{\Omega} dt \cdot \cos \left(\int_t \left(\frac{rW}{l_1^2} \frac{rW}{l_2^2} \frac{-rW}{l_3^2} \frac{-rW}{l_4^2} \right)^\top \cdot \dot{\Omega} dt \right) - \bar{x} \right)^2 \quad (71)$$

and

$$\sigma_y^2 = \frac{1}{N} \sum_{i=1}^N \left(\int_t (\lambda_1, \lambda_1, \lambda_1, \lambda_1)^\top \cdot \dot{\Omega} dt \cdot \sin \left(\int_t \left(\frac{rW}{l_1^2} \frac{rW}{l_2^2} \frac{-rW}{l_3^2} \frac{-rW}{l_4^2} \right)^\top \cdot \dot{\Omega} dt \right) - \bar{y} \right)^2, \quad (72)$$

as well as the robot's yaw statistical measurement model

$$\sigma_\theta^2 = \frac{1}{N} \sum_{i=1}^N \left(\int_t \left(\frac{rW}{l_1^2} \frac{rW}{l_2^2} \frac{-rW}{l_3^2} \frac{-rW}{l_4^2} \right)^\top \cdot \dot{\Omega} dt - \bar{\theta} \right)^2. \quad (73)$$

Furthermore, the measurement noise covariance matrix is

$$\mathbf{R}_k = \begin{pmatrix} \sigma_{x_p}^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_{y_p}^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_{y_p}^2 & 0 & 0 & 0 \end{pmatrix}, \quad (74)$$

and the matrix \mathbf{W}_k which is the partial derivative of the process model w.r.t. the process noise vector is

$$\mathbf{W}_k = \frac{\partial f}{\partial \mathbf{w}_k} = \begin{pmatrix} \frac{\partial x_k}{\partial w_x} & \frac{\partial x_k}{\partial w_y} & \frac{\partial x_k}{\partial w_\theta} \\ \frac{\partial y_k}{\partial w_x} & \frac{\partial y_k}{\partial w_y} & \frac{\partial y_k}{\partial w_\theta} \\ \frac{\partial \theta_k}{\partial w_x} & \frac{\partial \theta_k}{\partial w_y} & \frac{\partial \theta_k}{\partial w_\theta} \end{pmatrix}. \quad (75)$$

The matrix $\mathbf{V}_k = \partial h / \partial \mathbf{v}_k$ is the partial derivative w.r.t. the measurement noise vector,

$$\mathbf{V}_k = \begin{pmatrix} \frac{\partial x_k}{\partial v_{X_p}} & \frac{\partial x_k}{\partial v_{Y_p}} & \frac{\partial x_k}{\partial v_{Z_p}} \\ \frac{\partial y_k}{\partial v_{X_p}} & \frac{\partial y_k}{\partial v_{Y_p}} & \frac{\partial y_k}{\partial v_{Z_p}} \end{pmatrix}. \quad (76)$$

Let us summarize the 3D points \mathbf{p}_{AB} and \mathbf{p}_{BC} obtained by Theorem 1.

4.1. State feedback position control

This section describes in six general steps the combined use of the visual observers and the EKF geometric odometer as a recursive feedback for the robot's positioning control. The robot's deterministic kinematic model conveys predictions about the robot's geometry of motion and its observations. Therefore, the deterministic model is used to infer the robot's motion observations implicitly by the trinocular sensor. The following formulation illustrates how the EKF and the visual odometry model are fed back for the 4WD kinematics.

4.1.1. Kalman gain

The initial estimate of Kalman gain is

$$\mathbf{k}_k = \mathbf{P}_k \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k \mathbf{H}_k^T + \mathbf{V}_k \mathbf{R}_x \mathbf{V}_k^T)^{-1} \quad (77)$$

4.1.2. Observation

From Proposition 1.1, the visual observers provide m 3D key-points from Theorem 1, $\mathbf{p}_{AB,BC}$

$$\Delta s(\mathbf{p}) = \sqrt{\|\mathbf{p}_{t-1}\| + \|\mathbf{p}_t\| - 2\|\mathbf{p}_{t-1}\|\|\mathbf{p}_t\|\cos(\beta_{t-1} - \beta_t)}$$

The angle of each key-point \mathbf{p} or \mathbf{q} w.r.t. to the robot in actual time is

$$\lambda_t = \arcsin\left(\frac{\|\mathbf{p}_t\| \sin(\hat{\beta})}{\Delta s}\right),$$

and the local angle of the robot w.r.t. the robot's previous position is

$$\phi = \lambda_t + \left(\frac{\pi}{2} - \theta_{t-1}\right),$$

thus the inferred displacement is

$$\mathbf{x}_k = \begin{pmatrix} \frac{1}{m+n} \sum_i \Delta s_i^a(\mathbf{p}) \cos(\phi_i^a) \\ \frac{1}{m+n} \sum_i \Delta s_i^a(\mathbf{p}) \sin(\phi_i^a) \\ \lambda + \left(\frac{\pi}{2} - \theta_{t-1}^a\right) \end{pmatrix}. \quad (78)$$

Therefore, the observation vector with Gauss noise \mathbf{w} is

$$\mathbf{z}_k = \mathbf{H} \cdot \mathbf{x}_k + \begin{pmatrix} w_x \\ w_y \\ w_\theta \end{pmatrix}. \quad (79)$$

4.1.3. Update estimate

The update estimate is obtained by

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k-1}^{-} + \mathbf{K}_k(\mathbf{z}_k - \mathbf{H}\mathbf{x}_{k-1}). \quad (80)$$

4.1.4. Update error covariance

The covariance matrix error dispersion of the system is updated

$$\hat{\mathbf{P}}_k = \mathbf{P}_k - \mathbf{K}_k \mathbf{H}_k \mathbf{P}_k. \quad (81)$$

4.1.5. Deterministic control model

Therefore, the prediction is firstly obtained through the robot's inverse position control model, from the inverse kinematics equation, Eq. (38)

$$\boldsymbol{\Omega}_{t+1} = \boldsymbol{\Omega}_t + \boldsymbol{\Lambda}_t^{-1} \cdot (\mathbf{u}_R^{ref} - \hat{\mathbf{x}}_k),$$

where, in the previous expression, $\mathbf{u}_R = (s, \theta, x_R, y_R)^\top$. Thus, $\hat{\boldsymbol{\Omega}}_t = (\frac{2\pi}{R} \Delta\eta_1, \frac{2\pi}{R} \Delta\eta_2, \frac{2\pi}{R} \Delta\eta_3, \frac{2\pi}{R} \Delta\eta_4)^\top$ is the vector of the wheels' instantaneous measurements

$$\mathbf{u}_{t+1} = \mathbf{u}_t - \boldsymbol{\Lambda}_t^{-1} \cdot (\boldsymbol{\Omega}_{t+1} - \hat{\boldsymbol{\Omega}}_t).$$

This step converges until $(\mathbf{u}_R^{ref} - \hat{\mathbf{u}}_t) < \varepsilon_u$, where ε_u is the convergence error. Then, the robot's prediction model is

$$\bar{\mathbf{x}}_k = \bar{\mathbf{x}}_{k-1} + \mathbf{B} \cdot \mathbf{u}_{t+1}$$

\mathbf{B} being a control transition matrix.

4.1.6. State prediction

It follows that state prediction is

$$\bar{\mathbf{x}}_{k+1} = \boldsymbol{\Phi}_k + \bar{\mathbf{x}}_k + \mathbf{q}_k \quad (82)$$

and the error dispersion covariance matrix is also predicted at $t + 1$

$$\mathbf{P}_{k+1} = \mathbf{P}_k + (\mathbf{A} + \mathbf{A}^\top) \mathbf{P}_k \Delta t + (\mathbf{A} \mathbf{P}_k \mathbf{A}^\top - \Sigma_W) \Delta t^2 \quad (83)$$

From the previous step, the estimation process repeats again, going to step one. The previous Kalman process is performed until the robot reaches the goal and the estimation error converges by numerical approximation according to $(\mathbf{x}_k - \hat{\mathbf{x}}_k) \leq \varepsilon_x$.

Therefore, **Figure 4a** shows the robot's trajectory obtained by the different comparative approaches conducted in this study. The postures measured by an external visual global reference system are the main references to be compared with. The EKF estimation was obtained by the use of Theorem 1, Proposition 1.1, and Eqs. (71)–(77). In addition, the trinocular key-points used as inputs of the visual odometry model inferred the robot's displacements, which are shown in same **Figure 4a**. Furthermore, the dead-reckoning robot system was deployed to infer the robot's postures and is also shown in **Figure 4a**. Raw odometry refers to the robot's dead-reckoning kinematic model used as a mean for direct posture observation through direct kinematics (33) and inverse kinematics (38), but using direct encoder readings by (14).

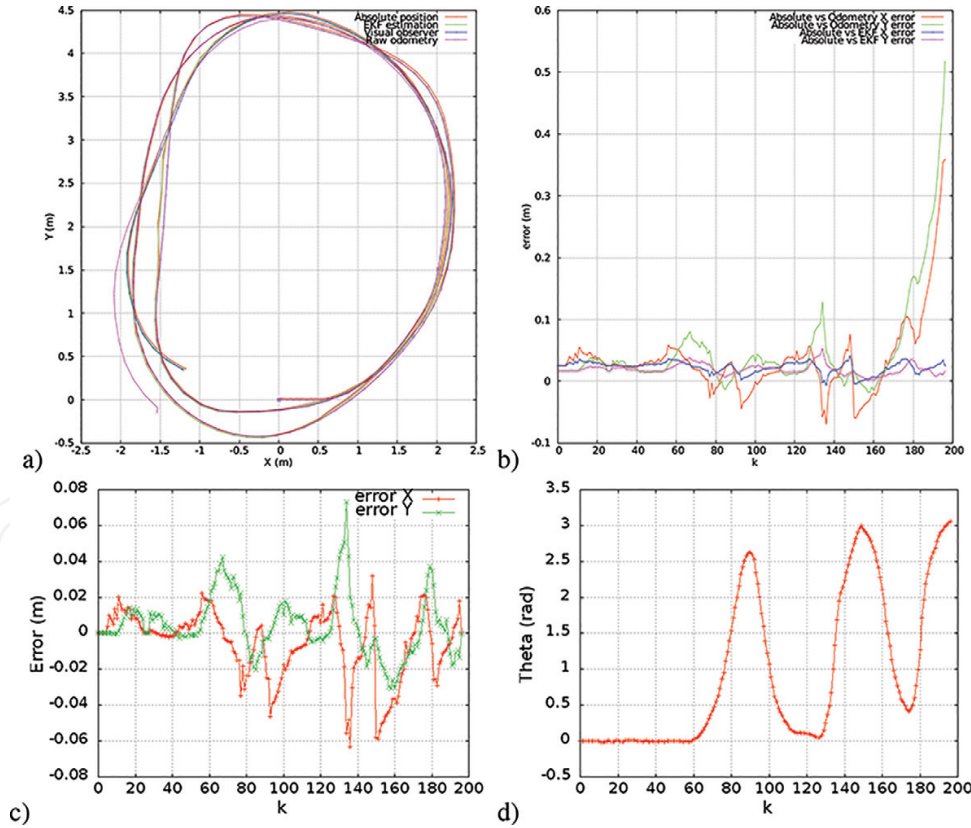


Figure 4. Positions and errors. (a) Cartesian positions. (b) Errors vs. global reference. (c) EKF's Cartesian errors' convergence. (d) EKF's angle error convergence.

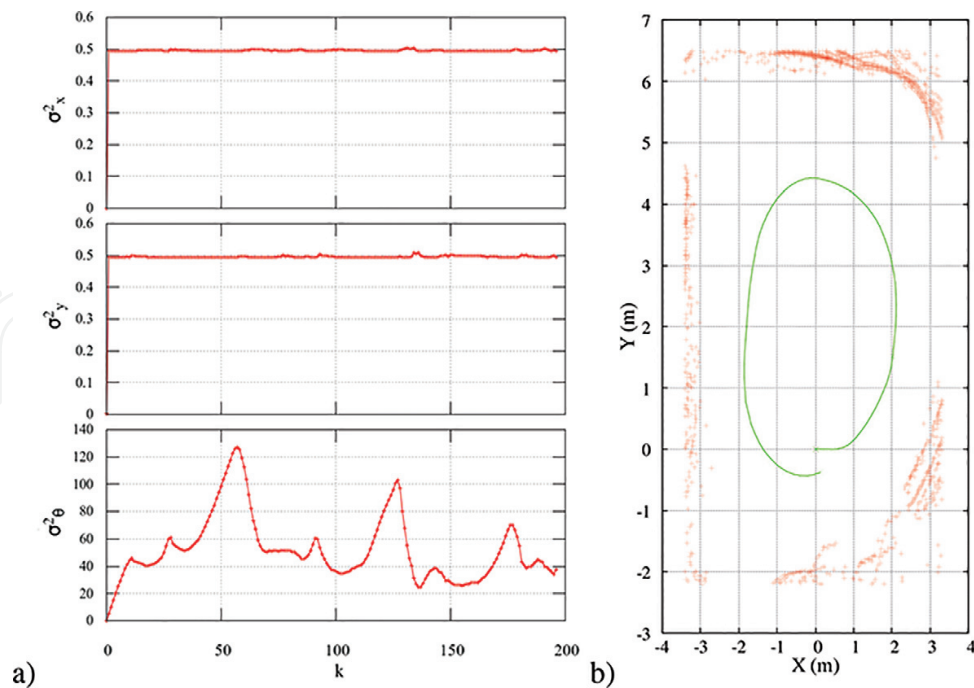


Figure 5. Errors' convergence behavior. (a) EKF variances over time. (b) Key-point map's divergence using state vectors.

Figure 4b shows the dead-reckoning and the EKF Cartesian absolute errors, taken as main reference for the visual global reference system. As for the direct dead-reckoning measurements, the absolute error grows exponentially, where the position observation starts diverging before the robot reaches the third turn. As for the EKF model, the Cartesian error w.r.t. the global reference does not diverge but preserves bounded error magnitudes.

As for **Figure 4c** and **d**, the EKF's Cartesian and angular absolute errors w.r.t. the global visual tracker are shown. In **Figure 4d**, the local minimums and maximums determine the Cartesian regions where the robot performed its turns.

Finally, **Figure 5a** shows the covariance error behavior obtained at each control loop during the EKF recursive calculations. **Figure 5b** is a mapping of the measured key-points registered using the state vector (posture) of a robot's turn to illustrate the map's divergence.

5. Conclusion

This chapter presented a visual odometry scheme for a trinocular divergent visual system that was combined with an EKF for visual odometry estimation. The proposed trinocular geometric model observer geometrically combined adjacent radial views. About 20% of adjacent multi-view overlapping data allowed inference of small volumes of depth information. In measuring 3D key-points, the X-axis metrical error was reported to be lower than 7 cm, with error less

than 10 cm in $+Y$ component and less than 3 cm in $-Y$ (vertical). Likewise, we found an averaged Z-axis error less than 5 cm (depth). Such errors were mostly produced by the angular divergence of the object w.r.t. the central camera, rather than linear distances. Indoor experiments, measuring distances up to 10 m were developed. In addition, a set of experimental results in convergent robot's course gave closed loops, and as the robot moved, the trinocular sensor incrementally stored environmental 3D key-points.

The robot's trajectory was obtained by different comparative approaches conducted in this study. The postures were measured by an external visual global reference system, which was the main reference system to be compared with. The robotic platform's kinematics was modeled in terms of a dead-reckoning approach. The direct and the inverse solutions were combined to produce a recursive linearized control model and this was used as the prediction model for EKF estimator. The dead-reckoning robot system was deployed to infer the robot's postures using directly the four encoders' readings, with good results obtained only for very short paths. As a comparative perspective, using only the 4WD dead-reckoning system the posture exponentially diverged.

We found bounded Cartesian error for this 4WD robot by deploying the EKF. The trinocular 3D key-points were used as inputs of the visual odometry model that inferred the robot's displacements by geometrical triangulations.

Author details

Edgar Alonso Martínez-García^{1*} and Luz Abril Torres-Méndez²

*Address all correspondence to: edmartin@uacj.mx

1 Laboratorio de Robótica, Universidad Autónoma de Ciudad Juárez, Mexico

2 Robotics Active Vision Group, CINVESTAV, Campus Saltillo, Mexico

References

- [1] Milella A, Reina G. Towards autonomous agriculture: Automatic ground detection using trinocular stereovision. *Sensors*. 2012;**12**(9):12405-12423
- [2] Gwo-Long L, Chi-Cheng C. Acquisition of translational motion by the parallel trinocular. *Information Sciences*. 2008;**178**:137-151
- [3] Tae Choi B, Kim J-H, Jin Chung M. Recursive estimation of motion and a scene model with a two-camera system of divergent view. *Pattern Recognition*. 2010;**43**(6):2265-2280

- [4] Jaeheon J, Correll N, Mulligan J. Trinocular visual odometry for divergent views with minimal overlap. In: IEEE Workshop on Robot Vision; 2013. pp. 229-236
- [5] Dellaert F, Kaess M. Probabilistic structure matching for visual slam with a multi-camera rig. *Computer Vision and Image Understanding*. 2010;**114**(2):286-296
- [6] Lee C-H, Kwon S, Lee J-h, Lim Y-C, Lee M. Improvement of stereo vision-based position and velocity estimation and tracking using a stripe-based disparity estimation and inverse perspective map-based extended kalman filter. *Optics and Lasers in Engineering*. 2010; **48**(9):859-886
- [7] Harmat A, Wang DWL, Sharf I, Waslander SL, Tribou MJ. Multi-camera parallel tracking and mapping with non-overlapping fields of view. *The International Journal of Robotics Research*. 2015;**34**(12):1480-1500
- [8] Seet G, Wei M, Han W. Efficient visual odometry estimation using stereo camera. In: 11th IEEE International Conference on Control and Automation; 2014. pp. 1399-1403
- [9] Kanade T, Badino H, Yamamoto A. Visual odometry by multi-frame feature integration. In: IEEE International Conference on Computer Vision Workshops; December 2013. pp. 222-229
- [10] Liang Z, Xu X. Improved visual odometry method for matching 3D point cloud data. In: 33rd Chinese Control Conference; July 2014; pp. 8474-8478
- [11] Llorca DF, Parra I, Sotelo MA, Ocaña M. Robust visual odometry for vehicle localization in urban environments. *Robotica*. 2010;**28**:441-452
- [12] Rendon-Mancha JM, Fuentes-Pacheco J, Ruiz-Ascencio J. Visual simultaneous localization and mapping: A survey. *Artificial Intelligence Review*. 2015;**43**(1):5581
- [13] Neira J, Cisneros R, Lavion JE, Hernandez E, Ibarra JM. Visual slam with oriented landmarks and partial odometry. In: 21st International Conference on Electrical Communications and Computers; February–March 2011. pp. 39-45
- [14] Hong S, Ye C. A pose graph based visual slam algorithm for robot pose estimation. In: World Automation Congress; August 2014. pp. 917-922
- [15] Williams B, Reid I. On combining visual slam and visual odometry. In: IEEE International Conference on Robotics and Automation; May 2010. pp. 3494-3500
- [16] Huitl R, Schroth G, Kranz M, Steinbach E, Hilsenbeck S, Moller A. Scale-preserving long-term visual odometry for indoor navigation. In: International Conference on Indoor Positioning and Indoor Navigation; November 2012. p. 110
- [17] Martínez-García EA, Lerin E, Torres-Cordoba R. A multi-configuration kinematic model for active drive/steer four-wheel robot structures. *Robotica*. 2015;**33**:2309-2329. Cambridge University Press

- [18] Martinez-Garcia E, Mar A, Torres-Cordoba T. Dead-reckoning inverse and direct kinematic solution of a 4W independent driven rover. In: IEEE ANDESCON; 15–17 September 2010; Bogota, Colombia
- [19] Martinez-Garcia EA, Torres-Cordoba T. 4WD skid-steer trajectory control of a rover with spring-based suspension analysis. In: International Conference in Intelligent Robotics and Applications (ICIRA2010); 10–12 November 2010; Shanghai, China