Robotica (2022), **xx**:xx 1–7
doi:10.1017/xxxxx

CAMBRIDGE
UNIVERSITY PRESS

RESEARCH ARTICLE

NMPC Based Visual Path Following Control with Variable Perception Horizon

Tiago T. Ribeiro*[1], Iago José P. B. Franco[1] and André Gustavo S. Conceição[1]

[1]LaR - Robotics Laboratory, Department of Electrical and Computer Engineering, Federal University of Bahia, Salvador, Bahia, Brazil.
*Corresponding author. E-mail: tiagotr@ufba.br.

Received: xx xxx xxx; **Revised:** xx xxx xxx; **Accepted:** xx xxx xxx

Keywords: Vision-based Control, Robot Navigation, Nonlinear Model Predictive Control, Computer Vision, Path Following, Optimal Control.

Abstract

For greater autonomy of visual control-based solutions, especially applied to mobile robots, it is necessary to consider the existence of unevenness in the navigation surface, an intrinsic characteristic of several real applications. In general, depth information is essential for navigating three-dimensional environments and for the consistent parameter calibration of the visual models. This work proposes a new solution, including depth information in the Visual Path-Following (VPF) problem, which allows the variation of the perception horizon at runtime while forcing the coupling between optical and geometric quantities. A new NMPC (Nonlinear Model Predictive Control) framework considering the addition of a new input to an original solution for the constrained VPF-NMPC allows the maintenance of low computational complexity. Experimental results in an outdoor environment with a medium-sized commercial robot demonstrate the correctness of the proposal.

1. Introduction

Growing computational power associated with high-performance embedded instrumentation has led to autonomous vehicles being susceptible to applications in the most varied contexts. Whether for safety or efficiency reasons, it is necessary to increase the autonomy levels in several applications, such as intelligent transport systems ([1], [2]), search and rescue ([3], [4]) or for navigation in industrial environments compatible with the precepts of Industry 4.0 ([5], [6]), to name a few.

Especially for these applications, the wide availability of low-cost visual sensors has enabled the development of controllers capable of generating control actions directly from the image plane, such as the classical IBVS (Image-Based Visual Servoing) based solutions. For this class of controllers, recent solutions range from classical probabilistic methods ([7]) to deep reinforcement learning ([8]) approaches.

Among the strategies for robot control based on computer vision, there are relevant techniques for visual path-following, capable of regulating a robot along an arbitrary visual path acquired at runtime. In this case, there are significant advances for swimming robots ([9]), humanoids ([10]), UAVs (Unmanned Aerial Vehicles) ([11]), or even surgical robots ([12]), with the citations limited to the last three years only.

The constrained nature of the camera's field of view, in addition to the peculiar characteristics of luminosity and frame rate, encourage the application of optimal controllers, with NMPC (Nonlinear Model Predictive Control) being an ideal strategy, due to its ability to apply directly to nonlinear, multivariable and constrained models, in addition to its good inherent robustness characteristics ([13]).

Despite the easy adaptation to the requirements of different application scenarios, this class of controllers has been the subject of constant investigations regarding computational cost analysis [14], and new formulations focused on efficiency [15, 16].

Although visual path-following methods produce good results for planar paths, as proposed by [17] and [18], even associated with NMPC as in [19], and [20], in several real situations, such as autonomous navigation on highways or factory floors of mezzanine format, unevenness in the navigation surface deserve special attention, as the reference paths are non-planar. In this case, path-following solutions must consider the problem's three-dimensionality to keep the visual system's calibration parameters coherent. This reality justifies the recent effort in evaluating strategies for lane line detection, such as the systematic review carried out in [21].

An immediate solution to this problem is the use of in-depth information. However, in order not to increase the computational complexity of the proposals, it is necessary to objectively define what information is relevant since the complete treatment of a point cloud for the estimation of a three-dimensional path, as would be the case of applying formal techniques for semantic segmentation of the path [22, 23] increases the dimension of the problem, making it impossible to use optimal and interactive controllers such as the NMPC.

Implementing this class of controllers in the solution of path-following problems has particular computational load requirements due to the prediction and constrained non-linear optimization resolution stages, which, when added to the demands of sophisticated stages of computer vision, substantially compromise the available processing bandwidth. In this context, alternatives arise based on analytical learning [24] through disturbance modeling by Gaussian processes, the human-like concept of visual memory [25], deep reinforcement learning [26], among others that directly address the problem of path following of mobile robots with greater computational efficiency since applications do not always allows high embedded computational power.

This article proposes a new solution for navigation along non-planar paths by including RGB-D sensors, which provide depth information at specific points for generating control actions directly from the image plane. Starting from the original model proposed by [27] and improved by [28], a degree of freedom is used to define the visual horizon and, for the maintenance of low computational complexity indices, the new scheme adds a new input to the NMPC algorithm, forcing the coupling between optical and geometrical quantities. The main advantage of this proposal is due to its ability to explicitly handle constraints, being able to control the pose of the robot along a visual reference path, through an optimal perception horizon, even on irregular and uneven navigation surfaces, among other imperfections.

To the best of the authors' knowledge, it is the first work that deals with the problem of optimal control under constraints for following visual paths in uneven outdoor terrain with a view to low computational cost through an effective load balance between the computer vision and control stages. It is an effective and generic solution that can be applied individually in structured environments with physical visual paths or in conjunction with a superior perception layer capable of providing virtual visual paths.

Implementation results through the ROS framework, using the robot Clearpath Husky UGV (Unmanned Ground Vehicle) and the RGB-D sensor Microsoft Kinetic in an external navigation environment, demonstrate that the proposed method produces satisfactory results for navigation on non-planar surfaces.

The remainder of this article is structured as follows: Section 2 briefly formalizes the problem. Section 3 presents the proposed new model and the version of the NMPC algorithm used. Section 4 presents the results, and Section 5 the main conclusions.

2. Problem formalization

Figure 1 presents the necessary elements for geometrically modeling the problem of visual path following, as initially proposed by [27]. In this case, for a prespecified linear velocity profile v, the states (identical to the present case outputs) consist of features extracted from a computer vision system at each iteration.

Initially, having as a premise that the robot and the visual system are always in front and longitudinal to the reference path, it defines a Serret-Frenet $\{SF\}$ system at a point P_r, representing the movement of a virtual robot that navigates at a constant linear velocity v. Such a point is defined through a distance H that defines a visual horizon line[1] in front of the robot. It intersects the path at P_r perpendicularly.

The geometric states shown in Figure 1(a) are the lateral displacement Z and the error angle θ_r between the longitudinal line to the robot and the tangent line to the path at P_r. The visual quantities corresponding to these states are presented in Figure 1(d), where Δ_{x_i} is the visual correspondent for Z and Δ_α is the visual correspondent for θ_r. Such correspondences are defined through individual calibration constants, as detailed further in this section. The H parameter has a component in the camera's field of view directly related to Δ_{y_i} and is geometrically calculated as follows:

$$H = \frac{l}{2} + d_1 + d_2; \tag{1}$$

$$d_1 = h_c \tan\left(\theta_{cam} - \frac{\theta^v_{fov}}{2}\right); \tag{2}$$

$$d_2 = k_h \Delta y_i, \tag{3}$$

where:

- l: robot length;
- h_c: camera height;
- d_1: distance between the camera reference frame and the bottom point of the vertical field of view (out of the image plane);
- d_2: distance between the bottom coordinate of the vertical field of view and P_r in the image plane (inside the image plane);
- θ_{cam}: camera focal axis angle;
- θ^v_{fov}: vertical field of view angle;
- k_h: visual horizon calibration constant;
- y_i: pixels in the vertical direction in the image plane.

The original model (Model 1 hereafter) for visual path-following is given as follows:

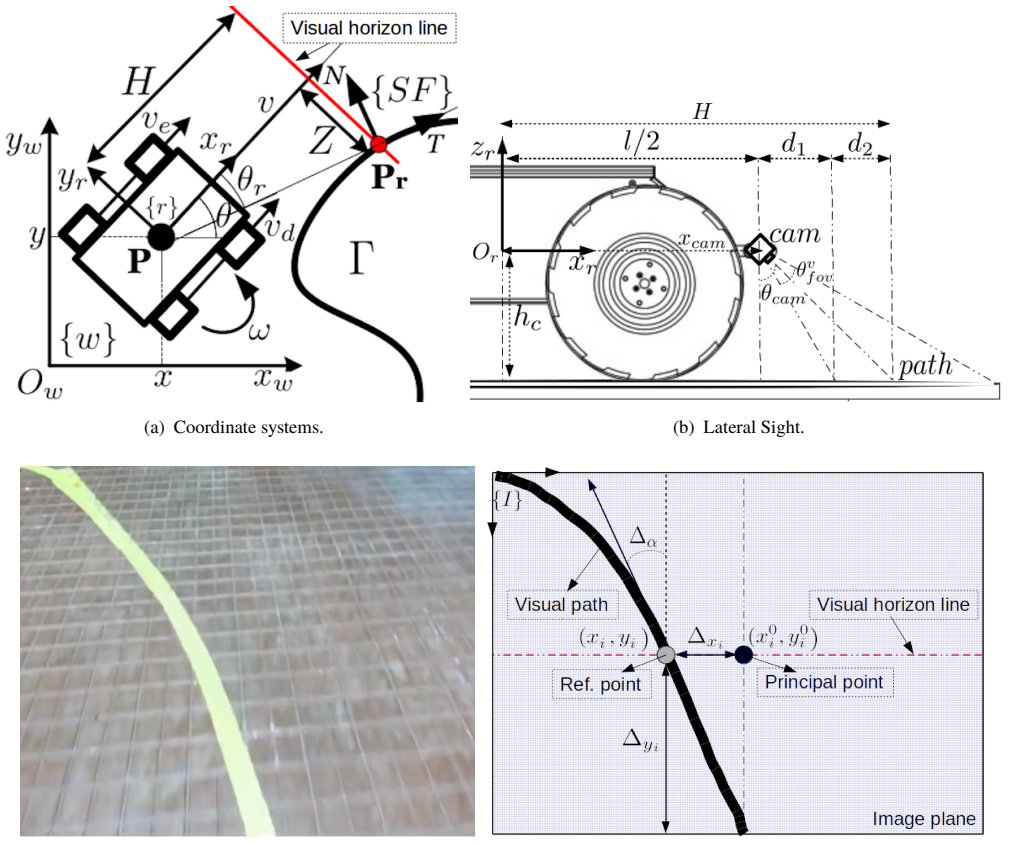
$$u_e = \omega - \omega_r, \tag{4}$$

$$\dot{\mathbf{x}}_e = \begin{bmatrix} \dot{Z} \\ \dot{\theta}_r \end{bmatrix} = \begin{bmatrix} \omega H + (\omega Z + v)\tan(\theta_r) \\ u_e \end{bmatrix}. \tag{5}$$

where:

- u_e: control input;
- ω: robot angular velocity;
- ω_r: virtual vehicle angular velocity;
- $\mathbf{x}_e$: two-dimensional state vector;

[1] This is an imaginary line, only for calculating the quantities of interest.



(a) Coordinate systems.

(b) Lateral Sight.

(c) Real image.

(d) Image plane.

Figure 1. *Modeling the visual path following problem.*

- s: Path length;
- $c(s)$: Path Curvature at s, given by $\frac{\omega_r}{\dot{s}}$;

Note that the underactuated nature of the system justifies the search for alternative models and controllers to guarantee performance metrics compatible with the most varied types of applications.

Despite good performance in controlled environments, the original proposal suffers from several practical problems. To solve issues with visual path discontinuity and curvature calculation, low ambient luminosity, among other imperfections of typical navigation scenarios of real applications, [28] proposed the interpolation of the visual path through a second-degree equation of the type $x_p = a_p {y_p}^2 + b_p y_p + c_p$, where x_p and y_p are coordinates of *pixels* in the image plane, providing to calculate the curvature as follows:

$$c = \frac{2a_p}{(1 + (2a_p y_p + b_p)^2)^{\frac{3}{2}}}. \tag{6}$$

The existence of a well-defined mathematical object for calculating the states and the curvature parameter allows us to follow paths with more complex curvature profiles. It makes it possible to propose new techniques considering the temporal variation of other parameters, such as the visual perception horizon proposed in the present work.

With this new way of estimating the visual path, we have an analytical method to obtain the current system states (Z and θ_r), given as follows:

$$Z = k_z \left(\frac{a_p}{k_h^2} d_2^2 + \frac{b_p}{k_h} d_2 + c_p - x_0 \right); \tag{7}$$

$$\theta_r = k_\theta \operatorname{atan} \left(\frac{k_h Z}{k_z d_2} \right). \tag{8}$$

where:

- $d_2 = H - \frac{l}{2} - d_1$ the component of the visual perception horizon in the image plane, as defined in (1);
- x_0: horizontal coordinate of the vertex of the second-order function on the image plane
- k_z: lateral displacement calibration constant;
- k_θ: angular error calibration constant.

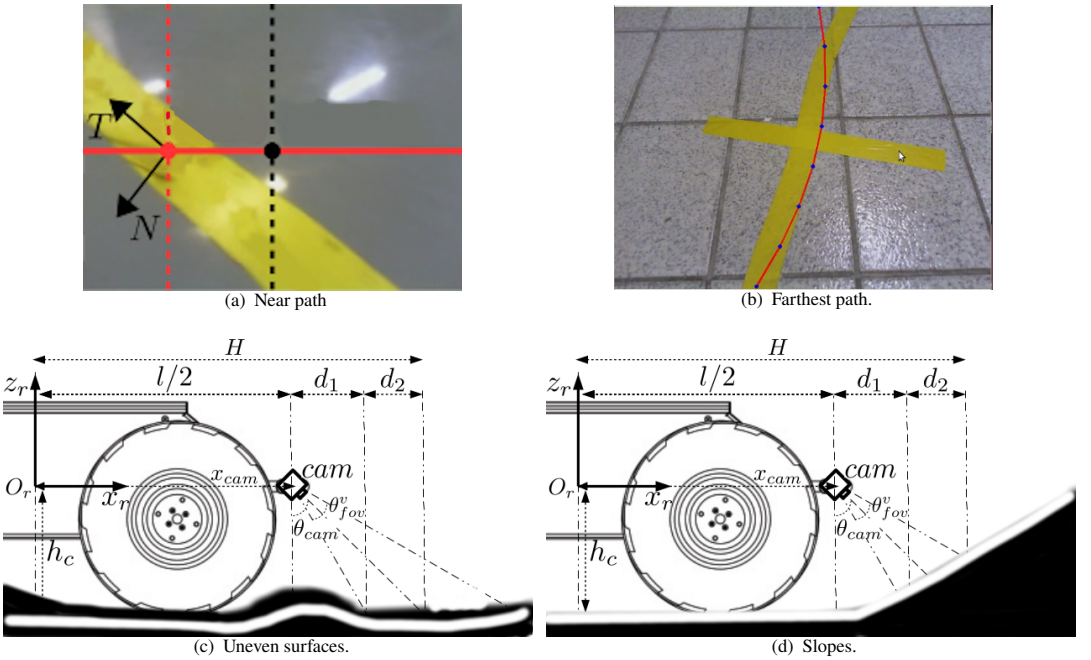
As y_p is related to H through d_2, there is a formal representation for the path, increasing the representation of the static visual path-following model. However, it still has limited applicability in some practical situations, as highlighted below:

- **Parameter calibration:** As can be seen in (7) and (8), both states depend on the calibration constant k_h, initially obtained by the relation between the number of pixels on the axis y_p of the image plane, for a single visual horizon value. Thus, it is necessary to define three calibration constants, which increases the uncertainties in the measurements of the states.
- **Constant horizon:** Assuming a constant visual horizon invalidates the use of Model 1 on uneven terrain, as it will generate inaccurate measurements of the states due to the slope of the terrain. Moreover, this model imposes limitations on model-based controllers since it employs the prediction horizon concept without establishing a physical correspondence with the visual horizon and misuses the receding horizon concept.
- **Model representativeness:** Taking the time derivatives of (7) and (8), a substantial kinematic inconsistency with (5) is observed (for $H = \text{cte} \rightarrow \dot{H} = \dot{Z} = \dot{\theta}_r = 0$), restricting the potential gains from the application of model-based techniques. This fact forces the visual quantities to be used only as an initial guess for the states, having no connection with the kinematic model obtained from the geometric relations.

For successfully implementing the Model 1 approach, whose path profile in the image plane is illustrated in Figure 2(a), it is necessary to position the camera in front of and very close to the path. Thus it is possible to reduce the effects of distortions in the images and obtain unique calibration constants. This constraint makes the path appear practically straight in all frames so that the accuracy of the measured curvatures is not critical. Interpolation using a second-order function, as shown in Figure 2(b), can compensate for imperfections along the path and extend the fixed horizon to higher values. Nevertheless, obtaining the calibration parameters remains a challenging task

Additionally, Figure 2 still illustrates two practical situations that justify the previous highlights. In Figure 2(c), the robot moves on an uneven surface, typical of natural outdoor environments, in such a way that parameter calibration is practically impossible without some preliminary information on the nature of the irregularities. In Fig 2(d), one can see that the robot cannot navigate on non-planar surfaces, given that the model used is incapable of considering variations in the horizon.

Finally, one of the most critical problems is the lack of correspondence between the kinematics of pixels in the image plane and the state variables in the real world since with H constant makes consistency only in a motionless case, since in this case (4) and (5) would become zero, which violates the assumption of a constant velocity profile v.



(a) Near path (b) Farthest path.

(c) Uneven surfaces. (d) Slopes.

Figure 2. *Typical challenges of real situations.*

In order to not further increase the computational complexity of the prior schemes, we solve this issue by proposing a new model, contemplating variations in the visual horizon through depth information, and exploring the good inherent robustness characteristics of NMPC controllers, as detailed in the next section.

3. NMPC-based Visual Path Following Control with Variable Perception Horizon

Aiming for simplicity and considering that low computing power is available, we propose using RGB-D cameras to acquire depth information at runtime. With such information, it is possible to obtain the visual horizon directly from the images and efficiently calculate the constant k_h through simple trigonometric relations, partially solving the issue relating to estimating visual parameters.

Additionally, it will be possible to modify the camera's pose so that the path in the image plane becomes more representative, enabling the identification of broader profiles through longer perception horizons. For this purpose, it is necessary to adjust the Model 1, starting from the geometric relations illustrated in Figure 1 and from the side views illustrated in Figures 2(c) and 2(d), as follow:

$$P_r(s(t)) = P(t) + H(t)\vec{x_r}(\theta(t)) - Z(t)\vec{y_r}(\theta(t)). \tag{9}$$

From the time derivative of the previous expression:

$$\begin{aligned} &\dot{s_T}\vec{T}(s) + \dot{s_N}\vec{N}(s) = \\ &= \dot{x}\vec{x_r}(\theta(t)) + \dot{y}\vec{y_r}(\theta(t)) + \dot{H}\vec{x_r}(\theta(t)) + H(t)\dot{\theta}\vec{y_r}(\theta(t)) + \\ &+ Z(t)\dot{\theta}\vec{x_r}(\theta(t)) - \dot{Z}\vec{y_r}(\theta(t)). \end{aligned} \tag{10}$$

Knowing that $\dot{s_T} = \dot{s}$ and $\dot{s_N} = \dot{y}\vec{y_r} = 0$ due to non-holonomic constraints, omitting the angular and temporal dependencies:

$$\dot{s}\vec{T}(s) = (\dot{x} + \dot{H} + Z\dot{\theta})\vec{x_r} + (H\dot{\theta} - \dot{Z})\vec{y_r}. \tag{11}$$

The relationship between the robot coordinate system $\{r\}$ and the *Serret-Frenet* $\{SF\}$ system is given as follows:

$$\begin{bmatrix} \vec{x_r} \\ \vec{y_r} \end{bmatrix} = \begin{bmatrix} \cos\theta_r & \sin\theta_r \\ -\sin\theta_r & \cos\theta_r \end{bmatrix} \begin{bmatrix} \vec{T} \\ \vec{N} \end{bmatrix}. \tag{12}$$

Projecting this expression in the *Serret-Frenet* system and replacing the kinematic model of the differential drive ($\dot{x} = v\cos\theta; \dot{y} = v\sin\theta; \dot{\theta} = \omega$):

$$\dot{Z} = \omega H + (v + \dot{H} + \omega Z)\tan\theta_r; \tag{13}$$

$$\dot{s} = \frac{v + \dot{H} + \omega Z}{\cos\theta_r}. \tag{14}$$

Since $\dot{\theta_r} = \omega - \dot{s}c(s)$, we have:

$$\dot{Z} = \omega H + (v + \dot{H} + \omega Z)\tan(\theta_r); \tag{15}$$

$$\dot{\theta_r} = \omega - c(s)\frac{(v + \dot{H} + \omega Z)}{\cos\theta_r}. \tag{16}$$

The current development makes it possible to establish a direct relationship between the measurements of quantities in pixels, measured in the image plane, for the quantities measured in meters, obtained from geometric modeling. This fact is noticeable when observing that the time derivative of (7) and (8) are no longer null (Model 1 case) since $\dot{Z}$ and $\dot{\theta}$ are given as follows:

$$\dot{Z} = \left(2\frac{a_p}{k_h^2}d_2 + \frac{b_p}{k_h}\right)\dot{H}; \tag{17}$$

$$\dot{\theta_r} = \frac{k_\theta k_h k_z(\dot{Z}d_2 + Z\dot{H})}{(k_z d_2)^2 + (k_H Z)^2}. \tag{18}$$

As can be seen with the proposed model, $\dot{Z}$ and $\dot{\theta}$ are related to $\dot{H}$ through optical quantities and calibration parameters, providing greater representativeness of the model and coherence between the visual and geometric quantities.

For a preliminary analysis of the proposed model non-linearity, we get the equilibrium points of the system (13) and (14) as follows:

$$\theta_r = \sin^{-1}(-Hc); \tag{19}$$

$$Z = \frac{\omega\sqrt{1 - (Hc)^2}}{c} - (v + \dot{H}). \tag{20}$$

Thus, there are no equilibrium points for Hc outside the interval [-1,1] and a discontinuity in $c = 0$. Consequently, there is no trivial way to apply analytical linearization techniques directly and use traditional schemes to handle the stability of NMPC controllers. Also, it is possible to note the importance of H for the present model, even if restricted to instrumental aspects, since there is no direct way to control the curvature parameter.

Considering the availability of distance information to calculate the current H, we propose to use $\dot{H}$ as a degree of freedom for changes in the visual horizon by adding a new input to the NMPC algorithm as follows:

$$u_1 = \frac{\dot{H}}{\cos\theta_r}. \tag{21}$$

Another control action, referring to angular velocity errors, is maintained like Model 1 approach, that is:

$$u_2 = \omega - c(s)\frac{(v + \omega Z)}{\cos\theta_r}. \tag{22}$$

The new model (Model 2 hereafter) for the visual path-following problem, considering variations in the perception horizon and compatible with the non-linear representations in state space, typically used in the application of predictive controllers, is finally written as follows:

$$\mathbf{u}_e = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \tag{23}$$

$$\dot{\mathbf{x}}_e = \begin{bmatrix} \dot{Z} \\ \dot{\theta}_r \end{bmatrix} = \begin{bmatrix} \omega H + \left(\frac{\omega - u_2}{c(s)}\right)\sin\theta_r + u_1\sin\theta_r \\ u_2 - c(s)u_1 \end{bmatrix}. \tag{24}$$

Considering that the outputs are the states themselves, the problem of following visual paths, with variable perception horizon, for differential robots can be summarized as follows:

Find $\dot{H}$ and ω, such that u_1, u_2 Z and θ_r are feasible.

With this proposal, it is possible to use depth information at specific points of interest without requiring a complete point cloud treatment to estimate a three-dimensional path, thus maintaining low computational cost requirements.

It is worth mentioning that the approach proposed here can be directly applied to structured environments, such as those commonly found for AGVs (Automated Guided Vehicles) navigation on the factory floor or docking stations in general. A prominent case can be found in [29], where controllers for VPF based on NMPC were responsible for essential navigation tasks in a case study of additive manufacturing operations by a mobile manipulator within a practical application[2].

However, the path does not necessarily need to be physical. It is possible to explicit the generality of the proposal by considering that the physical path can be used only in a training step and then removed or even eliminated by using a virtual path that can be generated at runtime by a layer added on top of the proposals in this article.

[2]More details at http://www.fastenmanufacturing.eu/index.php/visual-path-following-control/

3.1. NMPC Control scheme

The model represented by (23) and (24) is nonlinear, time-varying, and has constraints on inputs and states (outputs), justifying the use of computationally efficient optimal control strategies. Predictive control-based approaches meet some requirements due to their performance with constrained, time-varying, multivariable problems. Due to the moving horizon principle, such controllers have good inherent robustness characteristics and adapt well to disturbances, nonlinearities, and modeling errors. In order to obtain effective solutions for the regulation of states around the origin ($Z = \theta_r = 0$) with low computational complexity requirements, this article deals with the following continuous-time NMPC approach:

$$J_{min} = \min_{\mathbf{u}_e} \int_{t}^{t+T_p} F(\mathbf{x}_e(\tau), \mathbf{u}_e(\tau))d\tau, \tag{25}$$

$$\text{subject to: } \dot{\mathbf{x}}_e(\tau) = f(\mathbf{x}_e(\tau), \mathbf{u}_e(\tau)), \tag{26}$$

$$\mathbf{u}_e(\tau) \in \mathcal{U}, \forall\, \tau \in [t, t + T_c], \tag{27}$$

$$\mathbf{x}_e(\tau) \in \mathcal{X}, \forall\, \tau \in [t, t + T_p], \tag{28}$$

with the stage cost F given by:

$$F(\mathbf{x}_e(\tau), \mathbf{u}_e(\tau)) = \mathbf{x}_e^T \mathbf{Q} \mathbf{x}_e + \mathbf{u}_e^T \mathbf{R} \mathbf{u}_e, \tag{29}$$

where:
T_p: Prediction horizon;
T_c: Control horizon; With $T_c \leq T_p$;
$\mathcal{U}$: Set of feasible Inputs;
$\mathcal{X}$: Set of feasible states;
$\mathbf{Q}$, $\mathbf{R}$: Positive definite matrices that weight deviations from required values.

Due to the characteristics of the proposed model and the need to evaluate the proposal in comparison with the original method (based on Model 1), this work does not address techniques to guarantee feasibility or stability. Thus, the robustness characteristics inherent to NMPC controllers are explored by including a direct correspondence between the prediction and perception horizons.

After solving the optimization problem referring to the NMPC algorithm ((25) to (29)), as the final implementation step, the visual reference horizon H_{ref}, for the definition of $P_r(s(t))$ along the visual path, and the physical control effort ω_{ref}, are obtained using the optimal control inputs, $u_{1_{\text{opt}}}$ and $u_{2_{\text{opt}}}$, as follows:

$$H_{ref} = \int_{t=t_k}^{t=t_k+T_s} u_1(t)_{\text{opt}} \cos(\theta_r(t))du_1(t); \tag{30}$$

$$\omega_{ref} = \frac{u_2(t_k)_{\text{opt}} \cos\theta_r(t_k) + c(s)v}{\cos\theta_r(t_k) - cZ(t_k)}, \tag{31}$$

where t_k the actual sampling instant.

The Algorithm 1 provides a pseudocode of the proposed solution to characterize the proposal better. After that, the physical control actions (v, ω_{ref}) are sent to the internal control loop embedded in the robot (PID for the wheels), and the new reference visual horizon is updated for the calculation of new

features. Figure 3 illustrates more details of the elements necessary for the implementation. In this figure, the expressions implemented in each block are highlighted.

This new visual control method directly from the image plane makes it possible to navigate on irregular and non-planar surfaces, in addition to increasing the levels of robustness concerning imperfections in the visual system, as demonstrated in the experimental results of the next section.

Algorithm 1: Pseudocode of the Visual Path Following with variable horizon

Data: I_{RGB}, I_D ; /* Gets RGB and Depth Images */
Result: H_{ref}, ω; /* Provide optimal Horizon and Angular velocity */
while *true* **do**
  ; /* with $N \in \mathbb{N}^+$ and T_s the sampling instant */
  if $t = t_k = NT_s$ **then**
    $[I^*, H_{ref}] = fit\_curve(I_{RGB}, I_D)$; /* fit a second order fuction to image; detailed in: [28] */
    $[Z_0, \theta_{r_0}, u_{1_0}, u_{2_0}] = get\_features(I^*, H_{ref})$; /* get the initial states and control inputs initial guess from (7) and (8) */
    $[Z_{1..T_P}, \theta_{r_{1..T_P}}, u_{1_{0..T_{P-1}}}, u_{2_{0..T_{P-1}}}] = preditor(Z_0, \theta_{r_0}, u_{1_0}, u_{2_0})$; /* states propagation by truncated Taylor series based on (23) and (24) */
    $[u_{1_{opt}}, u_{2_{opt}}] = optimizer(Z_{1..T_P}\ \theta_{r_{1..T_P}}, u_{1_{0..T_{P-1}}}, u_{2_{0..T_{P-1}}})$; /* nonlinear optimization by SQP algorithm; detailed in: [30] */
    $[H_{ref}, \omega_{ref}] = get\_actions(u_{1_{opt}}, u_{2_{opt}})$; /* calculates the optimal physical control actions with (30) and (31) */
  end
end

4. Results

Initially, the proposed model was validated in a realistic simulation environment, built using the software *gazebo*, as seen in the video available at https://youtu.be/Ob7pSZ3O7_Q. These simulations show that only the method based on Model 2 can follow the path completely, enabling the assembly of a setup to acquire practical results.

For experimental evaluation, we define a scenario composed of a reference path with an arbitrary curvature profile, in yellow color, drawn along a non-planar navigation surface, as illustrated in Figure 4(a). The algorithms were developed using the *ROS framework* and applied to the *Clearpath Husky UGV* robot, illustrated in Figures 4(b) and 4(c), equipped with a sensor RGB-D *Microsoft kinect* with the following pose in relation to the robot's center of mass: x_{cam} = 0.4m; y_{cam} = 0m; z_{cam} = 0.5m; $\theta_{cam} = \pi/4$ rad.

The embedded computer system where the proposal is implemented as the following specifications: intel® CORE® i5 vPro 7th Gen, 8GB RAM, Ubuntu 16.04 LTS.

For an adequate comparison between the models, variations were made in the reference path to meet the cases of navigation between two different levels and curves in unevenness. Additionally, the evaluation criteria consider a scenario with significant variability of ambient lighting and long paths, aiming to show, in a practical sense, that the proposed technique is robust to errors in the acquisition of visual parameters and invariant to path length.

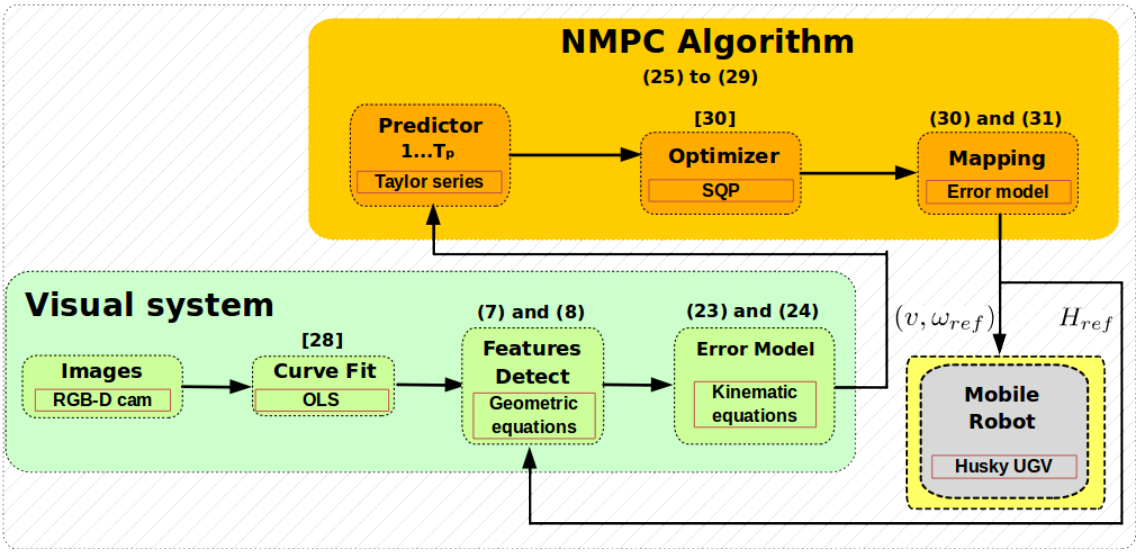
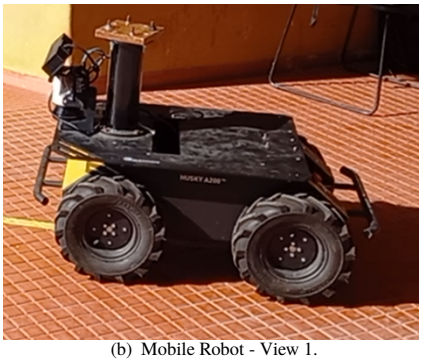


Figure 3. *Proposed scheme for NMPC-based visual path-following control with variable perception horizon. OLS: Ordinary Least Squares - The method used to fit the second-order curve to the path image (further details in [28]); SQP: Sequential Quadratic Programming - Non-linear optimization method used (further details in [30]).*

(a) Path on non-planar surface.



(b) Mobile Robot - View 1.

(c) Mobile Robot - View 2.

Figure 4. *Experimental environment.*

For the implementation of the NMPC controller, we consider $T_p = T_c = 3T_s$ due to the non-linearities mentioned above, and $T_s = 0.2$s, due to the dynamics of the open-loop system. The optimization problem was solved using the general-purpose non-linear optimizer DONLP2 ([30]).

In both Model 1 and 2 evaluations, Table 1 presents the tuning parameters, and Table 2 shows the constraints on states and inputs.

Table 1. *NMPC Tuning parameters.*

	Q	**R**
Model 1	$diag(0.1; 0.1)$	0.001
Model 2	$diag(0.1; 0.1)$	$diag(1; 0.01)$

Table 2. *NMPC inequality constraints.*

	$\|Z[m]\|$	$\|\theta_r[rad]\|$	$u_1[rad/s]$	$u_2[m/s]$
Model 1	0.5	0.5	2	-
Model 2	0.5	0.5	0.05	2

As it can be seen, the tuning parameter related to variations in the visual horizon has greater weight. The idea is to avoid high variations that would lead the controller to produce a horizon estimate outside the reference path due to imperfections in the visual system used for the test environment. This behavior is also specified via a low value for the constraint corresponding to this parameter.

The following subsections provide the main results obtained.[3]

4.1. Models comparison

A high curvature loop was added to the path in a high slope area for the present evaluation, as illustrated in Figure 5. The curved path is approximately 10m long and connects two uneven environments with a vertical distance of approximately 1.6m. We consider the start plan as level 0 and an artificial light environment.

The objective here is to directly confront the methods based on Models 1 and 2 for a navigation velocity of 0.3m/s, producing the results illustrated in Figure 6. It is observed in Figure 6(a)[4] that the method based on Model 1 loses the visual path in the section with high curvature and unevenness, around 25s. In contrast, the proposed method follows the entire path, regardless of its curvature or unevenness.

Figure 6(b) shows the runtime instantaneous curvature profiles, which are pretty noisy due to the performance of the visual parameters acquisition system for the scenario in question. As they are very similar profiles, we credited the impossibility of varying the perception horizon as the main reason for the poor performance of the method based on Model 1, confirming the existence of practical limits for the inherent robustness of NMPC controllers.

Figure 6(c) shows the measures of visual horizon variation given by $\Delta H = H(k) - H(k-1)$, measured in centimeters. It is possible to verify values coherent with the surface's unevenness both with the robot facing the unevenness and, laterally, at the point of most significant curvature.

Figure 6(d) shows the state errors, from which one can verify that the state constraints were satisfied throughout the experiment, even in the most critical stretch, where the errors came close to the maximum value due to the parameters of tuning used. Figure 6(e) shows control actions compatible with

[3] Videos with the results presented here are available in the supplementary materials.

[4] The color bar on the right side represents the measure of the unevenness along an axis transverse to the world plane.

Figure 5. *Environment for comparison between models.*

the experimental platform, reserve capacity for regulating more significant disturbances, and coherent activity of the proposed controller.

The following subsection investigates why Model 1 does not work in the present scenario more closely.

4.2. Limitation of Model 1

Calibration parameters were defined to level 0, related to the quota at the beginning of the experiment. Thus, it is natural to expect the method based on Model 1 to work correctly for just this case. To show this, Figure 7 illustrates the loop closing the path between the starting and ending points of the previous experiment. The idea is to evaluate the behavior of model 1 when returning to the level at which the parameters were originally calibrated.

Figure 8 illustrates the results obtained for a navigation velocity of 0.2m/s, a value even lower than in the previous experiment, aiming to increase the regulation capacity of this method. As expected, the robot gets lost when the curve is on the inclined surface, which it does not, even with degraded performance, at the flat level, where the approach has been calibrated to work.

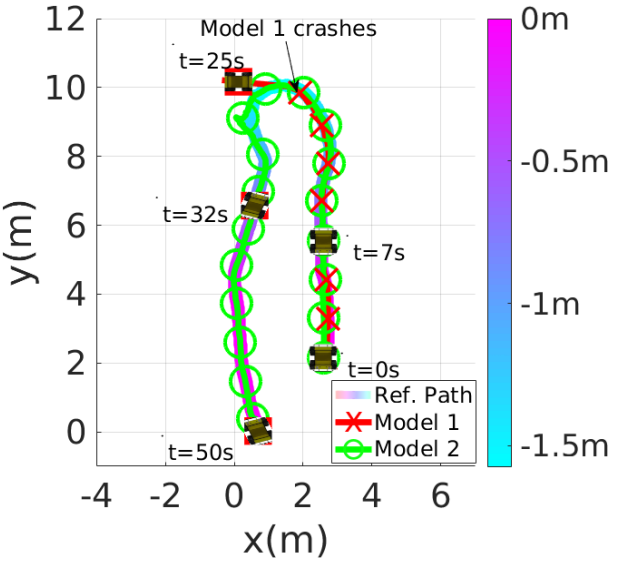
Figure 8(b) shows correctly regulations within limits established for the constraints, with an abrupt loss occurring when passing through the stretch of the greater curvature. In these results, we perform a manual rotation to repose the robot after losing its visual reference path in the image plane. From figure 8(c), one can see that it was not due to the platform's movement physical limitation since the control actions are far low from the maximum practicable. Thus, we confirm that the model has a significant limitation for this navigation scenario.

On the other hand, we confirmed the proposal's validity by running this same path for three different velocities, all of which the path was followed thoroughly. Without loss of generality to more straightforward and better direct analyses, we get two quantitative metrics, more specifically, the integral of absolute error (*IAE*) and the total control variation (*TV*). The *IAE* index, calculated by $\int_0^{T_{END}} |e(t)|dt$, is widely used to compare the performance of different strategies in similar experiments on the other hand, the *TV* index, calculated by $\sum_{k=0}^{k_{END}} |u(k)-u(k-1)|$, aims to evaluate the effect of noise on control signals. Table 3 presents the results.

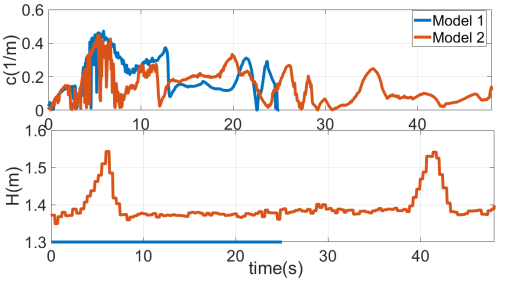
As can be seen, the parameters remained coherent and compatible with the specifications of the experimental platform in all cases.

v_{nav}[m/s]	IAE_Z[m]	IAE_{θ_r}[rad]	TV_v[m/s]	TV_ω[rad/s]
0.3	1.4914	2.2777	0.3	6.5415
0.5	2.1075	3.2305	0.5	10.0321
0.7	2.5687	3.6943	0.7	18.3398

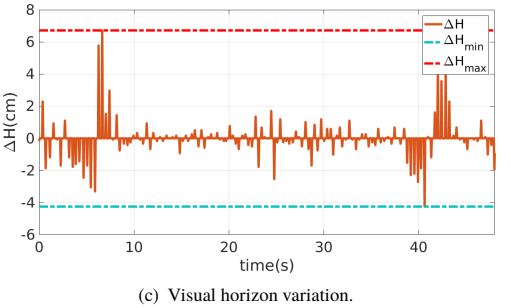
Table 3. *Three velocity quantitative comparison.*



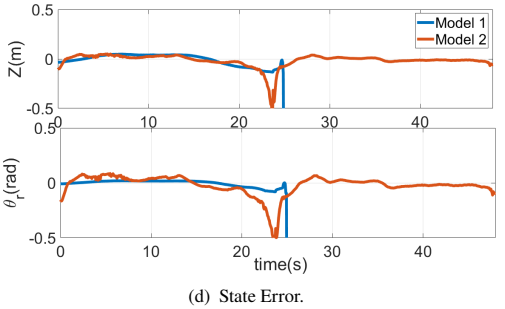
(a) Poses plane - Red crosses: Model 1 Result; Green circles: Model 2 Result.



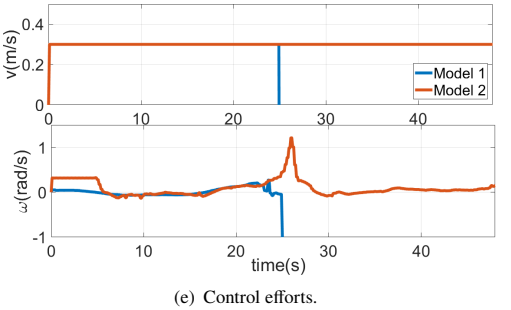
(b) Model parameters.



(c) Visual horizon variation.



(d) State Error.



(e) Control efforts.

Figure 6. *Models 1 and 2 based methods experimental results.*

4.3. Detailed analysis of Model 2

This section provides a detailed evaluation of the proposed method's performance. We define a long path arranged on an uneven surface, with significant variability of ambient lighting, in stretches with bifurcations and high curvature, as illustrated in Figure 9. The curved path is approximately 15m long and connects two uneven environments in approximately 3.8m. Additionally, an object was positioned in the transition zone between the lowest level and the ascent ramp, subjecting the proposed method to an even more aggressive disturbance.

The results obtained from the complete experiment, at $v = 0.2$ m/s, are illustrated in Figure 10. It is possible to note that the path was followed entirely, in both directions (down and up), even with the various imperfections added (see Figure 10(a)). Figure 10(b) shows the physical parameters acquired at runtime. As can be seen, even with intentional perturbations that make the equilibrium points undefined, the system can regulate the state errors, as shown in Figure 10(d).

Figure 10(c) shows the instant visual horizon variation measures. It is possible to notice minor variations due to the lower speed and an abrupt disturbance due to the object positioned along the path. The physical control actions are fully compatible with the experimental platform, as shown in Figure 10(e). Also, for this experiment, the instantaneous values of the internal control actions are shown in Figure 10(f), confirming that the inputs' constraints were fully satisfied.

Regarding computational performance, the instantaneous processing times were acquired throughout the experiment, as illustrated in Figure 10(g). It is possible to notice the computational efficiency of the proposal since, in just one sample, the processing time T_{proc} was close to the sampling period T_s, highlighting an average of approximately 10% of T_s.

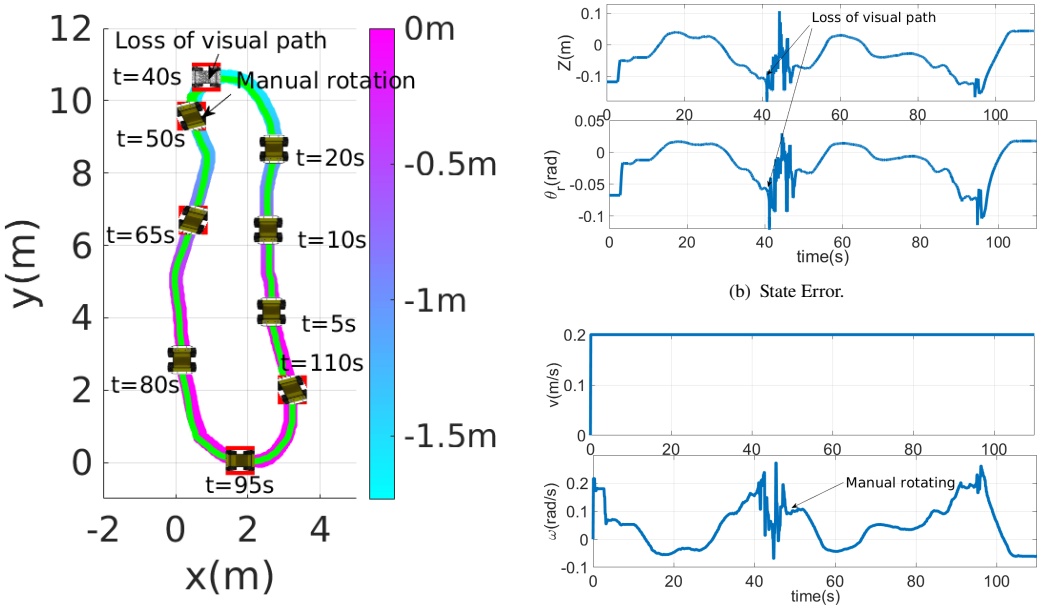
Finally, to attest to the repeatability of the proposal, the same experiment was performed three more times, producing the results in Figure 11. One can note similar and satisfactory performances, especially when considering a robot of about 50 kg weighing navigates a long and uneven path based only on visual information.

Tables 4 and 5 gather some statistical data obtained through these experiments. It is worth highlighting mean values for the error states and the physical control action close to zero, low standard deviations, including for the quantitative metrics IAE and TV.

$\overline{Z}$**[m]**	σ_Z**[m]**	$\overline{\theta_r}$**[rad]**	σ_{θ_r}**[rad]**	$\overline{IAE_Z}$**[m]**	σ_{IAE_Z}**[m]**	$\overline{IAE_{\theta_r}}$**[rad]**	$\sigma_{IAE_{\theta_r}}$**[rad]**
-0.0041	0.0314	-0.0061	0.0460	4.2468	2.5435	6.9137	3.9525

Table 4. *Three experiments state data analysis..*

Figure 7. *Environment for Model 1 analysis..*



(a) Poses plane.

(b) State Error.

(c) Control efforts.

Figure 8. *Model 1 analysis.*

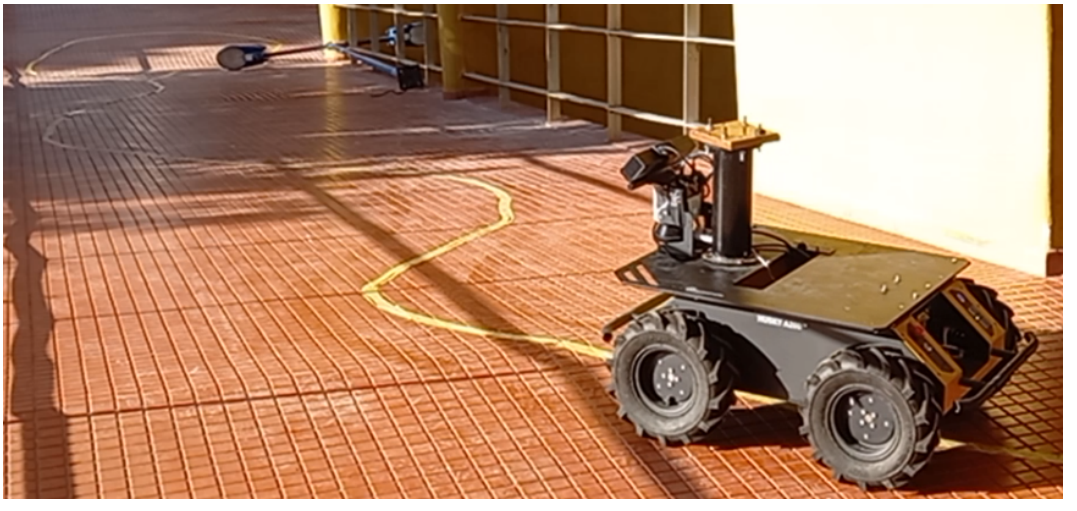
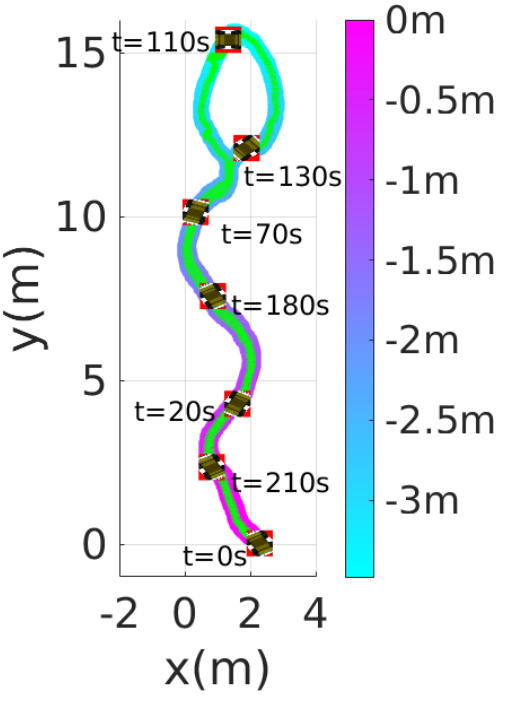


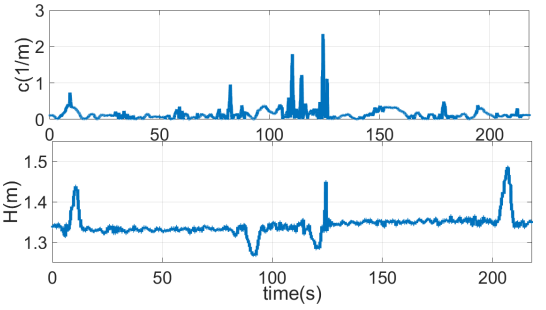
Figure 9. *Environment for Model 2 analysis.*

$\overline{v}$**[m/s]**	σ_v**[m/s]**	$\overline{\omega}$**[rad/s]**	σ_ω**[rad/s]**	$\overline{TV_v}$**[m/s]**	σ_{TV_v}**[m/s]**	$\overline{TV_\omega}$**[rad/s]**	σ_{TV_ω}**[rad/s]**
0.2	0.1067	-0.0012	0.0793	0.2	0.005	24.6209	16.8627

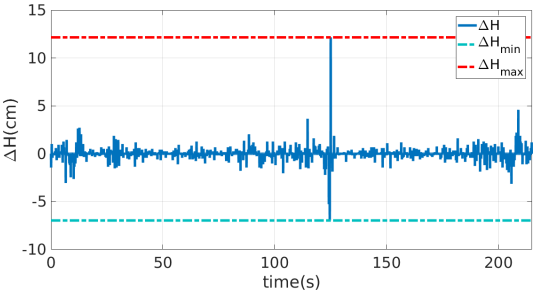
Table 5. *Three experiments control actions data analysis..*



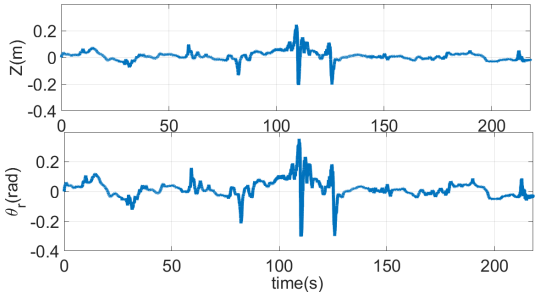
(a) Poses plane.



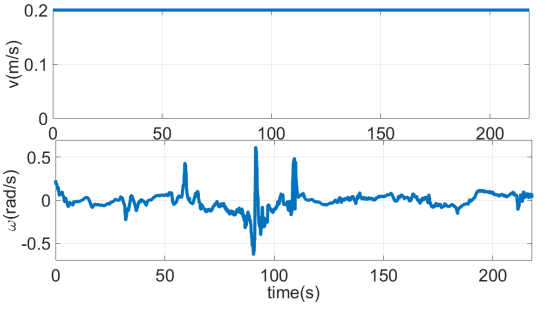
(b) Model parameters.



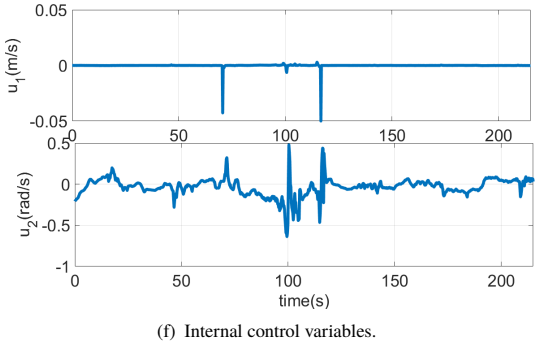
(c) Visual horizon variation.



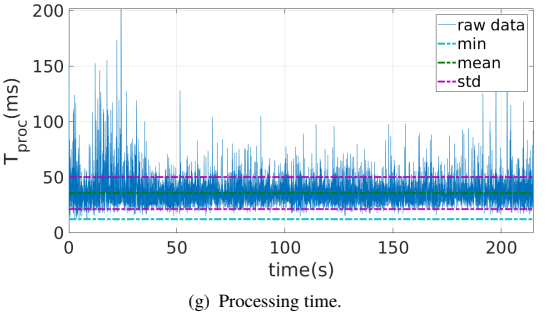
(d) State error.



(e) Control efforts.

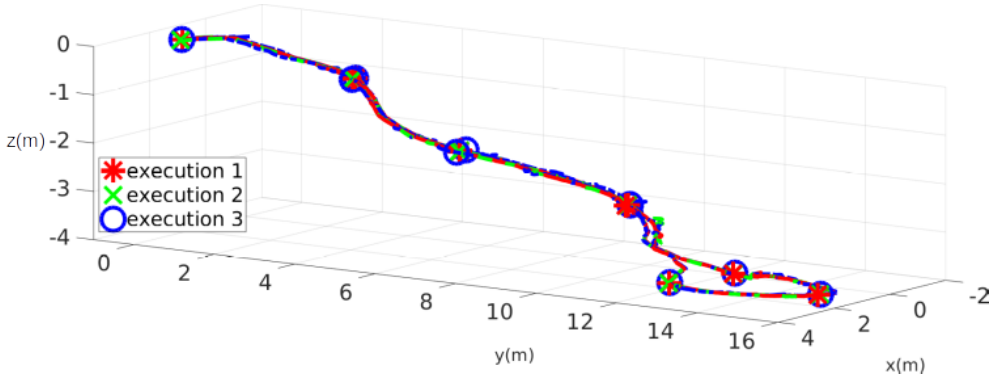


(f) Internal control variables.

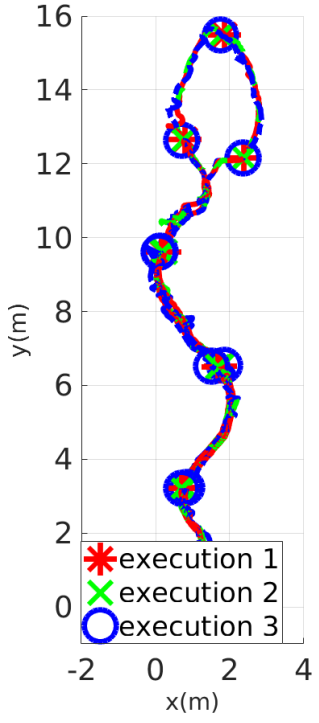


(g) Processing time.

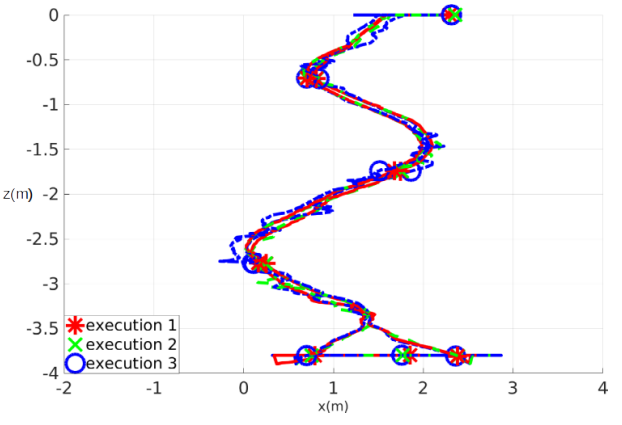
Figure 10. *Model 2 analysis.*



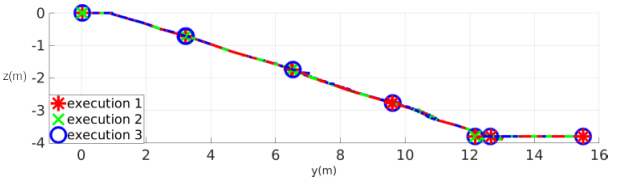
(a) Comparizon 3D view.



(b) Comparizon XY view.



(c) Comparizon XZ view.



(d) Comparizon YZ view.

Figure 11. *Three experiments comparison..*

5. Conclusions

This article proposes a new model for the visual perception horizon variation of NMPC-based visual path-following control. With this new model, navigation problems on uneven and non-planar surfaces are solved, in addition to calibrating visual parameters and ensuring consistency between optical and geometric quantities by including RGB-D sensors, enabling the extension of real applications.

To maintain the requirements of low computational complexity, we include specific and local depth information and define a degree of freedom used for the variation of the visual horizon, even if limited to the physical field of view, so that we detect the optimal point for parameter acquisition at runtime.

The inherent characteristics of the NMPC algorithm allow the application of the new proposed model, even without the explicit treatment of feasibility and stability, which could interfere with the overall computational cost of the strategy.

Experimental results in an outdoor navigation environment, using a commercial robot and visual sensor, demonstrated that, with the proposed approach, it is possible to follow visual paths in several adverse situations safely and efficiently, even on non-planar surfaces. These results also show the validity of proposals based on exploring the inherent robustness levels of NMPC controllers, to the detriment of sophisticated solutions, with high theoretical load in analytical terms or computationally costly implementation.

Future work includes investigating strategies for explicit horizon variation depending on a specific application, detailed analysis and treatment of feasibility and stability, and an explicit metric to forward velocity variation.

Author Contributions. Tiago T. Ribeiro - Developmens, Simulation, Experiments and algorithm validation, Writing, Reviewing and editing of manuscript. Iago José P. B. Franco - Discussion, Writing, Reviewing and editing of manuscript. André Gustavo S. Conceição - Supervision, Project Administration, Discussion, Experiments, Reviewing and editing of manuscript.

Financial Support. We would like to thank the SEPIN/MCTI and the European Union's Horizon 2020 Research and Innovation Programme through the Grant Agreement No. 777096, and the Brazilian funding agency (CNPq) Grant Numbers [311029/2020-5 and 407163/2022-0] and the CAPES - Finance Code 001.

Conflicts of Interest. The authors declare that they have no conflict of interest.

Ethical Approval. This article does not contain any studies with human participants or animals performed by any of the authors

References

[1] Raksha Ghosh, R Pragathi, S Ullas, and Surekha Borra. Intelligent transportation systems: A survey. In *2017 International Conference on Circuits, Controls, and Communications (CCUBE)*, pages 160–165, 2017.

[2] Fenghua Zhu, Yisheng Lv, Yuanyuan Chen, Xiao Wang, Gang Xiong, and Fei-Yue Wang. Parallel transportation systems: Toward iot-enabled smart urban traffic control and management. *IEEE Transactions on Intelligent Transportation Systems*, 21(10):4063–4071, 2020.

[3] Ahmad Baranzadeh and Andrey V. Savkin. A distributed control algorithm for area search by a multi-robot team. *Robotica*, 35(6):1452–1472, 2017.

[4] Hongling Wang, Chengjin Zhang, Yong Song, Bao Pang, and Guangyuan Zhang. Three-dimensional reconstruction based on visual slam of mobile robot in search and rescue disaster scenarios. *Robotica*, 38(2):350–373, 2020.

[5] Felipe S. Costa, Silvia M. Nassar, Sergio Gusmeroli, Ralph Schultz, André G. S. Conceição, Miguel Xavier, Fabiano Hessel, and Mario A. R. Dantas. Fasten iiot: An open real-time platform for vertical, horizontal and end-to-end integration. *Sensors*, 20(19), 2020.

[6] Majid Yekkehfallah, Ming Yang, Zhiao Cai, Liang Li, and Chuanxiang Wang. Accurate 3d localization using rgb-tof camera and imu for industrial mobile robots. *Robotica*, 39(10):1816–1833, 2021.

[7] Ruolong Qi, Yuangui Tang, and Ke Zhang. An optimal visual servo trajectory planning method for manipulators based on system nondeterministic model. *Robotica*, 40(6):1665–1681, 2022.

[8] Zhehao Jin, Jinhui Wu, Andong Liu, Wen-An Zhang, and Li Yu. Policy-based deep reinforcement learning for visual servoing control of mobile robots with visibility constraints. *IEEE Transactions on Industrial Electronics*, 69(2):1898–1908, 2022.

[9] Chenyang Huang, Tiantian Xu, Jia Liu, Laliphat Manamanchaiyaporn, and Xinyu Wu. Visual servoing of miniature magnetic film swimming robots for 3-d arbitrary path following. *IEEE Robotics and Automation Letters*, 4(4):4185–4191, 2019.

[10] Noé G. Aldana-Murillo, Luis Sandoval, Jean-Bernard Hayet, Claudia Esteves, and Hector M. Becerra. Coupling humanoid walking pattern generation and visual constraint feedback for pose-regulation and visual path-following. *Robotics and Autonomous Systems*, 128:103497, 2020.

[11] Eder A. Rodriguez Martinez, Guillaume Caron, Claude Pégard, and David Lara-Alabazares. Photometric-planner for visual path following. *IEEE Sensors Journal*, 21(10):11310–11317, 2021.

[12] Yan Wang, Qiyuan Sun, Zhenzhong Liu, and Lin Gu. Visual detection and tracking algorithms for minimally invasive surgical instruments: A comprehensive review of the state-of-the-art. *Robotics and Autonomous Systems*, page 103945, 2021.

[13] Douglas A. Allan, Cuyler N. Bates, Michael J. Risbeck, and James B. Rawlings. On the inherent robustness of optimal and suboptimal nonlinear mpc. *Systems & Control Letters*, 106:68–78, 2017.

[14] Ricus Husmann and Harald Aschemann. Comparison and benchmarking of nmpc for swing-up and side-stepping of an inverted pendulum with underlying velocity control. *IFAC-PapersOnLine*, 54(14):263–268, 2021. 3rd IFAC Conference on Modelling, Identification and Control of Nonlinear Systems MICNON 2021.

[15] Keerthi Chacko, Janardhanan Sivaramakrishnan, and Indra Kar. Computationally efficient nonlinear mpc for discrete system with disturbances. *International Journal of Control, Automation and Systems*, 20:1–10, 04 2022.

[16] Jan Reinhold, Henry Baumann, and Thomas Meurer. Constrained-differential-kinematics-decomposition-based nmpc for online manipulator control with low computational costs. *Robotics*, 12(1):7, Jan 2023.

[17] Albert Diosi, Anthony Remazeilles, Siniga Segvic, and Francois Chaumette. Outdoor visual path following experiments. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4265–4270, 2007.

[18] Feknous Safia and Chouireb Fatima. Visual path following by an omnidirectional mobile robot using 2d visual servoing. In *2017 5th International Conference on Electrical Engineering - Boumerdes (ICEE-B)*, pages 1–7, 2017.

[19] Guoxing Bai, Li Liu, Yu Meng, Weidong Luo, Qing Gu, and Baoquan Ma. Path tracking of mining vehicles based on nonlinear model predictive control. *Applied Sciences*, 9(7), 2019.

[20] Sorin Grigorescu, Cosmin Ginerica, Mihai Zaha, Gigel Macesanu, and Bogdan Trasnea. Lvd-nmpc: A learning-based vision dynamics approach to nonlinear model predictive control for autonomous vehicles. *International Journal of Advanced Robotic Systems*, 18(3):17298814211019544, 2021.

[21] Noor Jannah Zakaria, Mohd Ibrahim Shapiai, Rasli Abd Ghani, Mohd Najib Mohd Yassin, Mohd Zamri Ibrahim, and Nurbaiti Wahid. Lane detection in autonomous vehicles: A systematic review. *IEEE Access*, 11:3729–3765, 2023.

[22] Y. Hu, Z. Chen, and W. Lin. Rgb-d semantic segmentation: A review. In *2018 IEEE International Conference on Multimedia & amp; Expo Workshops (ICMEW)*, pages 1–6, Los Alamitos, CA, USA, jul 2018. IEEE Computer Society.

[23] Yang Zhang, Yang Yang, Chenyun Xiong, Guodong Sun, and Yanwen Guo. Attention-based dual supervised decoder for rgbd semantic segmentation. *ArXiv*, abs/2201.01427, 2022.

[24] Chris J. Ostafew, Angela P. Schoellig, and Timothy D. Barfoot. Robust constrained learning-based nmpc enabling reliable mobile robot path tracking. *The International Journal of Robotics Research*, 35(13):1547–1563, 2016.

[25] Ashish Kumar, Saurabh Gupta, David Fouhey, Sergey Levine, and Jitendra Malik. Visual memory for robust path following. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 765–774.

Curran Associates, Inc., 2018.

[26] Alan Maldonado-Ramirez, Reyes Rios-Cabrera, and Ismael Lopez-Juarez. A visual path-following learning approach for industrial robots using drl. *Robotics and Computer-Integrated Manufacturing*, 71:102130, 2021.

[27] Tiago T. Ribeiro and André G. S. Conceição. Nonlinear model predictive visual path following control to autonomous mobile robots. *Journal of Intelligent & Robotic Systems*, 95(2):731–743, Aug 2019.

[28] Iago José Pattas Bastos Franco, Tiago Trindade Ribeiro, and André Gustavo Scolari Conceição. A novel visual lane line detection system for a nmpc-based path following control scheme. *Journal of Intelligent & Robotic Systems*, 101(1):12, 2021.

[29] Rafael Arrais, Germano Veiga, Tiago T. Ribeiro, Daniel Oliveira, Ramon Fernandes, André Gustavo S. Conceição, and P. C. M. A. Farias. Application of the open scalable production system to machine tending of additive manufacturing operations by a mobile manipulator. In Paulo Moura Oliveira, Paulo Novais, and Luís Paulo Reis, editors, *Progress in Artificial Intelligence*, pages 345–356, Cham, 2019. Springer International Publishing.

[30] P. Spellucci. An sqp method for general nonlinear programs using only equality constrained subproblems. In *Mathematical Programming*, volume 82, pages 413 – 448, 1998.