Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 53-2 (2020) 9412–9419

A Perception-Aware Flatness-Based Model Predictive Controller for Fast Vision-Based Multirotor Flight [⋆]

Melissa Greeff [*] **Timothy D. Barfoot** [**] **Angela P. Schoellig** [***]

[*] *Dynamic Systems Lab, University of Toronto (e-mail: melissa.greeff@mail.utoronto.ca).*
[**] *Autonomous Space Robotics Lab, University of Toronto (e-mail: tim.barfoot@utoronto.ca).*
[***] *Dynamic Systems Lab, University of Toronto (e-mail: angela.schoellig@robotics.utias.utoronto.ca)*

Abstract: Despite the push toward fast, reliable vision-based multirotor flight, most vision-based navigation systems still rely on controllers that are perception-agnostic. Given that these controllers ignore their effect on the system's localisation capabilities, they can produce an action that allows vision-based localisation (and consequently navigation) to fail. In this paper, we present a perception-aware flatness-based model predictive controller (MPC) that accounts for its effect on visual localisation. To achieve perception awareness, we first develop a simple geometric model that uses over 12 km of flight data from two different environments (urban and rural) to associate visual landmarks with a probability of being successfully matched. In order to ensure localisation, we integrate this model as a chance constraint in our MPC such that we are probabilistically guaranteed that the number of successfully matched visual landmarks exceeds a minimum threshold. We show how to simplify the chance constraint to a nonlinear, deterministic constraint on the position of the multirotor. With desired speeds of 10 m/s, we demonstrate in simulation (based on real-world perception data) how our proposed perception-aware MPC is able to achieve faster flight while guaranteeing localisation compared to similar perception-agnostic controllers. We illustrate how our perception-aware MPC adapts the path constraint along the path based on the perception model by accounting for camera orientation, path error and location of the visual landmarks. The result is that repeating the same geometric path but with the camera facing in opposite directions can lead to different optimal paths flown.

Keywords: Vision-based Navigation, Model Predictive Control, Unmanned Aerial Vehicles

1. INTRODUCTION

In recent years, inspired in part by the DARPA Fast Lightweight Autonomy (FLA) challenge (DARPA (2015)) and the potential for autonomous drone racing (Kaufmann et al. (2018)), there has been a significant push toward fast, vision-based multirotor unmanned aerial vehicle (UAV) flight, see, for example, Mohta et al. (2018), Beul et al. (2018). One such vision-based approach uses a Visual Teach and Repeat (VT&R) framework that allows the UAV to repeat a previously taught path by matching current visual features to those in the locally metric map created during teach, see Gao et al. (2019), Warren et al. (2018).

Most of these vision-based navigation systems still rely on perception-agnostic control techniques. For example, a conventional model predictive control (MPC) computes a control input by optimizing a cost function over a prediction horizon but ignores the effect of this motion on the visual localisation capabilities of the system. This becomes especially relevant around sharp turns where there is often a trade-off between high speed and allowable path error that ensures localisation. An incorrect trade-off highly prioritising low path error leads to suboptimal slower flight while an incorrect trade-off highly prioritising fast flight can lead to a path error that results in a localisation failure by the visual system.

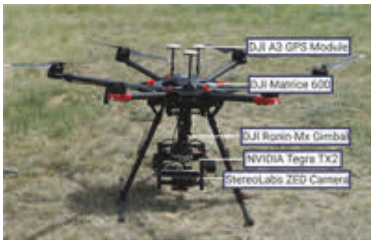


Fig. 1. Our experimental setup for Visual Teach & Repeat (VT&R) on a multirotor UAV.

This trade-off between fast and reliable flight is often addressed by adding a fixed allowable path error constraint, see Ostafew et al. (2016). However, because this trade-off is visual-environment dependent, these approaches do

[⋆] This research was supported by Drone Delivery Canada, Defence Research and Development Canada, and the Natural Sciences and Engineering Research Council of Canada.

10.1016/j.ifacol.2020.12.2411

not capture variability of this trade-off along a path and they have to be retuned in every new visual environment. Consequently, they tend to lead to suboptimal speeds or unreliable vision-based navigation.

Our aim is to implement a *perception-aware controller, that accounts for its effect on visual localisation in a VT&R framework, and to demonstrate its ability to achieve reliable but fast vision-based flight* compared to conventional perception-agnostic controllers.

2. RELATED WORK AND CONTRIBUTIONS

Similar to Falanga et al. (2018), we adopt a perception-aware MPC strategy. In Falanga et al. (2018), by using additional perception error terms in the MPC cost function, a multirotor UAV with static camera was able to follow a trajectory while tracking a single visual target. Unlike this approach, we are not limited by a multirotor with a static camera as we decouple the camera orientation from the multirotor UAV using a gimbal, see Warren et al. (2018). Our multirotor UAV platform with gimballed-stereo camera is shown in Fig. 1. Moreover, instead of a single target, there are features associated with many potential landmarks (observed during teach) that we could match and use to localise. A simple strategy of pointing the camera toward the centroid of these landmarks, see Patel et al. (2019), cannot overcome viewpoint changes as a result of UAV path error. This is because visual landmarks cannot be successfully matched when there is a significant perspective change. Another key difference, which avoids additional cost function tuning, is that we choose to treat the limitations of our visual perception as a constraint in the MPC formulation. The problem then becomes: *How do we successfully integrate our visual perception limits as a constraint into a real-time MPC?*

Similar to Churchill et al. (2015) and Gurau et al. (2017), we aim to model the limits of perception performance in a VT&R framework by generating an area around the teach path where localisation is probabilistically guaranteed (coined *localisation envelope* in Churchill et al. (2015)). Unlike these approaches, we close the control loop around vision by developing a perception model that is used in a constraint in real-time MPC. Unlike early work in Furgale et al. (2010), the authors in Churchill et al. (2015) developed a *localisation envelope* that could vary at different points along the teach path. They did this by modelling the likely number of feature matches around the teach path, where some low number of feature matches embodies the boundary of the *localisation envelope*. This is modelled using a Gaussian Process (GP), which takes as input the position relative to the path, the associated closest point on the teach path and the localiser's performance during multiple repeat runs. The authors propose repeat trials that deviate from the path until failure. This is both time consuming and potentially unsafe. In Gurau et al. (2016) and Gurau et al. (2017), the work in Churchill et al. (2015) is extended to similar teach paths by also utilizing curvature and an appearance model in the GP. Both approaches are limited by a place-dependent model to assess localisation performance. To overcome this limitation, our proposed approach instead creates a simple but conservative global *localisation envelope* model that does not have to be retrained for every new path. We do this by incorporating scene structure through associating landmarks seen in teach with a probability of being matched.

There are three areas of novelty to our proposed perception-aware MPC:

- We develop and validate a simple geometric perception model (for nominal lighting conditions - i.e., little difference in lighting conditions between teach and repeat) using over 12 km of data for our visual localisation system in Warren et al. (2018).
- We show how to integrate this perception model in a chance constraint, such that localisation is guaranteed, in our MPC and how to convert it to a deterministic nonlinear constraint.
- Using real-world perception data, we provide experimental simulation results demonstrating the value of our perception-aware MPC in terms of reliably but optimally self-regulating speed along a path compared to a similar perception-agnostic control.

3. BACKGROUND

3.1 Vision-Based Localisation

This section summarizes the vision-based localization framework used in this work. During the teach phase, the UAV flies using autonomous GPS waypoint following. The purpose of the teach path is to create a map of visual landmarks along the path. This taught path is stored as a set of vertices and edges which include the landmarks observed at each vertex.

During the repeat phase, GPS is disabled and the VT&R algorithm performs visual localisation using a local segment from the taught path. Specifically, at the current position (named leaf $\mathfrak{l}$), the localisation algorithm tries to match current observed landmarks with the stored 3-D landmarks from the spatially closest vertex (named trunk $\mathfrak{t}$) in the teach path. It does this by performing SURF feature matching and passing the raw feature matches through RANSAC to obtain a set of *localisation inliers*. These *localisation inliers* are used in a MLESAC robust estimator to estimate the relative transform from trunk to leaf $\mathbf{T}_{\mathfrak{l}\mathfrak{t}}$. This relative transform provides us with our position relative to the path which we use in our controller to repeat the path. Please refer to Warren et al. (2018) for more details.

If the number of *localisation inliers* drops below some threshold, a localisation failure occurs as a poor estimate of the relative transform will be obtained and we ultimately lose knowledge of the position of our multirotor which will prevent the control from repeating the path.

3.2 Multirotor Dynamics

We consider a cascaded control structure with a low-level onboard controller and an MPC outer-loop controller that sends commands $(\dot{z}_{cmd}, \phi_{cmd}, \theta_{cmd}, \dot{\psi}_{cmd})$, where $\dot{z}_{cmd}$ is the commanded vertical z-velocity, ϕ_{cmd} the commanded roll, θ_{cmd} the commanded pitch and $\dot{\psi}_{cmd}$ the commanded yaw rate.

We perform a simple system identification, as in Greeff et al. (2018), to approximate the inner-loop attitude dynamics by:

$$\dot{\phi} = \frac{1}{\tau_\phi}(k_\phi \phi_{cmd} - \phi), \tag{1a}$$

$$\dot{\theta} = \frac{1}{\tau_\theta}(k_\theta \theta_{cmd} - \theta), \tag{1b}$$

$$\dot{\psi} = \dot{\psi}_{cmd}, \tag{1c}$$

where τ_ϕ, τ_θ are identified time constants, k_ϕ, k_θ are identified gains and ϕ, θ, ψ are the roll, pitch and yaw angles of the vehicle. We perform a similar system identification to approximate the z-velocity dynamics by a second-order response:

$$\dddot{z} = -\frac{1}{\tau_z}\dot{z} - \frac{1}{\tau_{Iz}}\ddot{z} + \frac{1}{\tau_{Cz}}\dot{z}_{cmd}, \tag{2}$$

where $\tau_z, \tau_{Iz}, \tau_{Cz}$ are identified time constants. By ignoring drag and other external forces, we describe the lateral motion using the standard model, see Kamel et al. (2017):

$$\ddot{x} = \frac{\mathbf{R}_{13}}{\mathbf{R}_{33}}(\ddot{z} + g), \tag{3a}$$

$$\ddot{y} = \frac{\mathbf{R}_{23}}{\mathbf{R}_{33}}(\ddot{z} + g), \tag{3b}$$

where x, y, z represent the linear position, $\mathbf{R}$ the rotation of the multirotor with respect to an inertial frame and g the gravitational constant. We use the notation $\mathbf{R}_{13}$ to refer to the $(1, 3)$ entry of $\mathbf{R}$.

4. PROBLEM STATEMENT

Consider a multirotor with a continuous-time, nonlinear model of the form:

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t)), \quad \mathbf{x}(0) = \mathbf{x_0}, \quad t \in \mathbb{R}^+, \tag{4}$$

described by (1a)-(1c), (2) and (3a)-(3b) with state $\mathbf{x} = (x, y, z, \dot{x}, \dot{y}, \dot{z}, \ddot{z}, \phi, \theta, \psi)$, input $\mathbf{u} = (\dot{z}_{cmd}, \phi_{cmd}, \theta_{cmd}, \dot{\psi}_{cmd})$ and output:

$$\mathbf{y} = (x, y, z, \psi). \tag{5}$$

Given a teach path and some user-defined desired speed s_{des} by which to repeat the path, determine an optimal control problem (OCP) for real-time MPC that can be used to compute an input $\mathbf{u}(t)$ such that the following is achieved:

- We achieve high-speed flight, i.e., we track our desired speed such that $|s - s_{des}|$ remains small where $s = ||(\dot{x}, \dot{y}, \dot{z})||$.
- We guarantee that we can successfully localise using visual perception. In our VT&R navigational system, the success of our visual localisation during repeat is associated with the number of *localisation inliers*. When the number of localisation inliers L is below some threshold L_{min} we do not have a reliable estimate of our position. We, therefore, choose to represent the limitations of our visual perception by a perception failure chance constraint where the probability of the number of inliers going below our determined threshold remains very low. Specifically, we can write this perception constraint as:
$$\Pr(L < L_{min}) \leq \delta, \tag{6}$$
where δ is small.

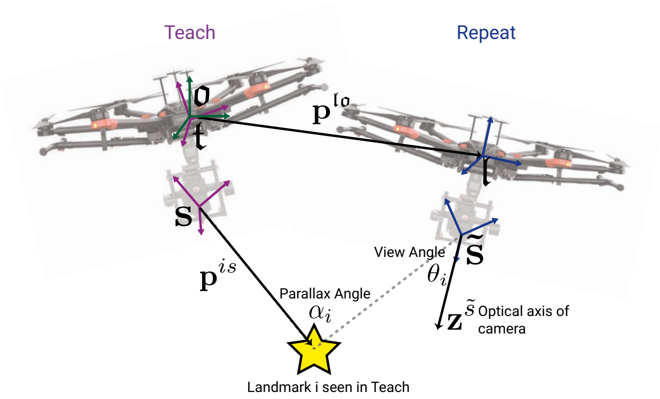


Fig. 2. Geometric angles used to model the probability p_i that a landmark i becomes a *localisation inlier*. *Parallax angle* α_i captures the perspective change as a result of path error between teach and repeat while *view angle* θ_i captures whether the landmark is still visible.

5. METHODOLOGY

We approach the problem in three steps:
1. Geometric Perception Model: By treating each landmark from the closest teach frame (trunk) independently, we develop a simple geometric model to associate a landmark with a probability of being matched successfully and becoming a localisation inlier. The model was identified using significant previous flight data (in nominal lighting conditions).
2. Perception Chance Constraint: Using this probability model, we transform our perception failure chance constraint from (6) into a deterministic but nonlinear constraint on the position of the multirotor UAV.
3. Perception-Aware Flatness-Based MPC: In a method similar to standard nonlinear MPC approaches, we linearize this constraint at each time step about our previously predicted trajectory and landmarks, and perform one iteration of a sequential quadratic program (SQP) online.

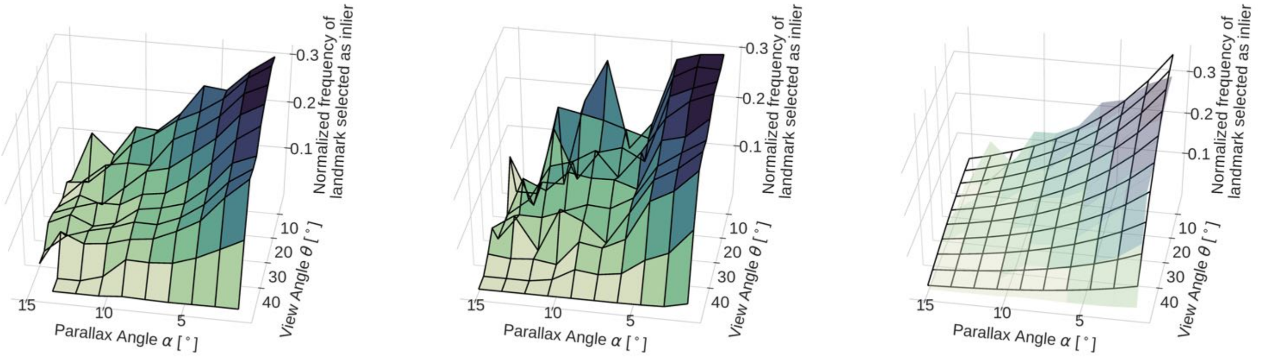
5.1 Geometric Perception Model

At the current position (leaf), we consider all landmarks from the trunk (closest in teach path) that we could potentially successfully match to become *localisation inliers*. Our aim is to associate each landmark i with a probability of being an inlier. Specifically, we treat each landmark i as an *independent Bernoulli random variable* l_i with some probability p_i of being an inlier, that is:

$$l_i \sim \text{Ber}(p_i). \tag{7}$$

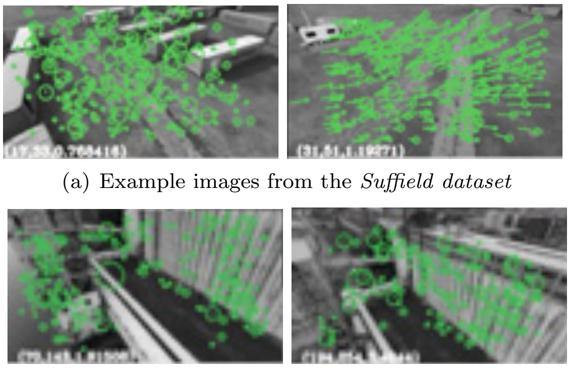
Our aim is to develop a simple perception model that can be used to estimate the probability p_i for each landmark i. Let $\mathbf{p}_{\mathfrak{o}}^{is}$ be the vector from the trunk sensor s to landmark i described in the inertial frame $\mathfrak{o}$. As shown in Fig. 2, we develop this perception model as a function of two geometric angles.

Parallax angle α_i is the angle between the rays from landmark i to the camera sensors s and $\tilde{s}$ as a result of the positional offset between the leaf and trunk. This angle captures the perspective change, i.e., how much the



(a) Suffield dataset (b) Montreal dataset (c) Best model fit on combined dataset

Fig. 3. Normalized frequency of landmark selected as a localisation inlier vs *parallax angle* α and *view angle* θ shown for (a) the *Suffield dataset*, (b) the *Montreal dataset* and (c) the combined dataset. In (c) a best fit model $c_1 \cos(c_2\theta)\exp(c_3\alpha)$ is overlayed in black where for optimal parameters $(c_1, c_2, c_3) = (0.424, 0.028, -0.145)$, a sufficient fit was achieved ($R^2 = 0.92$).



(a) Example images from the *Suffield dataset*

(b) Example images from the *Montreal dataset*

Fig. 4. Localisation inliers (green) associated with successfully matched landmarks between teach and repeat.

landmark has moved in the scene when reviewing it. It is computed as a function of the position offset $\mathbf{p}_{\mathfrak{o}}^{\mathfrak{l}\mathfrak{o}} = (x, y, z)$ (see Fig. 2) and is independent of the camera orientation:

$$\alpha_i = \angle(-\mathbf{p}_{\mathfrak{o}}^{is}, -\mathbf{p}_{\mathfrak{o}}^{is} + \mathbf{p}_{\mathfrak{o}}^{\mathfrak{l}\mathfrak{o}}). \tag{8}$$

View angle θ_i is the angle between the landmark i and the optical axis of the current sensor $\tilde{s}$. It captures both the effect of the field of view of the camera and potential degradation at the edges of the image. Let $\mathbf{z}_{\mathfrak{o}}^{\tilde{s}}$ be the vector representing the current sensor optical axis described in the inertial frame $\mathfrak{o}$. We can compute θ_i as:

$$\theta_i = \angle(\mathbf{z}_{\mathfrak{o}}^{\tilde{s}}, \mathbf{p}_{\mathfrak{o}}^{is} - \mathbf{p}_{\mathfrak{o}}^{\mathfrak{l}\mathfrak{o}}). \tag{9}$$

We use teach and repeat paths from two datasets to identify the perception model. *Montreal dataset*: We use 6 trials of a teach followed by a repeat in an urban environment near downtown Montreal, Canada. *Suffield dataset*: We use 6 trials of a teach followed by a repeat at a rural environment in Alberta, Canada. Fig. 4 shows examples of images observed. For all trials, for each landmark i in the trunk we compute the associated angles α_i and θ_i (as a result of position and camera offset) and we mark whether the landmark was selected as a *localisation inlier* or not. For each angle range (i.e. some α and θ), we compute the normalized frequency of a landmark being selected as an inlier by dividing the total number of inliers by the total number of landmarks seen. We show the results for the Suffield dataset, Montreal dataset, and then the combined dataset in Fig. 3. We consider a frequentist approach and associate the determined normalized frequency based on our data as a probability of that landmark being an inlier. We then fit the model:

$$p_i = c_1 \cos(c_2\theta_i)\exp(c_3\alpha_i) \tag{10}$$

where p_i is the probability of being an inlier, θ_i is the view angle and α_i is the parallax angle. The best least squares fit gives parameter constant values $(c_1, c_2, c_3) = (0.424, 0.028, -0.145)$. Using (8) and (9) in (10), we can rewrite the probability model as a nonlinear function of the output $\mathbf{y} = (\mathbf{p}_{\mathfrak{o}}^{\mathfrak{l}\mathfrak{o}}, \psi)$:

$$p_i = h(\mathbf{y}, \mathbf{p}_{\mathfrak{o}}^{is}, \mathbf{z}_{\mathfrak{o}}^{\tilde{s}}), \tag{11}$$

where $\mathbf{p}_{\mathfrak{o}}^{is}$ is determined from the known landmark position (determined during teach) and $\mathbf{z}_{\mathfrak{o}}^{\tilde{s}}$ is determined using the current gimbal orientation.

5.2 Perception Chance Constraint

We consider n landmarks at the trunk where we treat each landmark l_i as an independent non-identical Bernoulli trial given by (7). Each landmark is associated with a probability p_i based on (10) of being an inlier. Let L be the total number of successes of these trials (i.e. the total number of inliers) where L has a distribution associated with the sum of these n independent, non-identical Bernoulli trials. This distribution is known as a *Poisson binomial distribution* where its first two moments can be found by summing the expectation and variance of each non-identical Bernoulli trial respectively. This gives the expectation for L as:

$$\mathrm{E}(L|p_1, ..., p_n) = n\bar{p},$$

where $\bar{p} = \frac{1}{n}\sum_{i=1}^{n} p_i$ and the variance for L as:

$$\mathrm{Var}(L|p_1, ..., p_n) = n\bar{p}(1-\bar{p}) - ns_p^2,$$

where $ns_p^2 = \sum_{i=1}^{n}(p_i - \bar{p})^2 \geq 0$. Note from this that the expectation for L is an upper bound for its variance. Specifically,

$$\mathrm{Var}(L|p_1, ..., p_n) \leq n\bar{p}(1-\bar{p}) \leq n\bar{p} = \mathrm{E}(L|p_1, ..., p_n).$$

We tend to have many landmarks at the trunk (i.e. n tends to be around 600 to 800). By using *Lyapanov's Central Limit Theorem* (see Appendix A) as justification, we make a normal approximation for the Poisson binomial distribution L, that is $L \sim \mathcal{N}(\mu, \sigma^2)$, where $\mu = \mathrm{E}(L|p_1, ..., p_n)$ and $\sigma^2 = \mathrm{Var}(L|p_1, ..., p_n)$ as given by the expressions above. Using this normal approximation for L, we can now rewrite the visual perception failure chance constraint from (6):

$$\Pr(L < L_{min}) \leq \delta, \quad L \sim \mathcal{N}(\mu, \sigma^2)$$

as an equivalent deterministic constraint:

$$\mu - L_{min} \geq \bar{c}\sigma$$

where $\bar{c} = \sqrt{2}\mathrm{erf}^{-1}(1 - 2\delta)$ and erf^{-1} is the inverse error function. By using the upper bound for our variance, $\sigma^2 \leq \mu$, a more conservative constraint becomes:

$$\mu - \bar{c}\sqrt{\mu} \geq L_{min}. \tag{12}$$

By recalling that $\mu = \sum_{i=1}^{n} p_i$ and plugging in (11) we can rewrite this constraint as a nonlinear constraint on the output $\mathbf{y}$:

$$\sum_{i=1}^{n} h(\mathbf{y}, \mathbf{p}_{\mathfrak{o}}^{is}, \mathbf{z}_{\mathfrak{o}}^{\tilde{s}}) - \bar{c}\sqrt{\sum_{i=1}^{n} h(\mathbf{y}, \mathbf{p}_{\mathfrak{o}}^{is}, \mathbf{z}_{\mathfrak{o}}^{\tilde{s}})} \geq L_{min}. \tag{13}$$

5.3 Perception-Aware Flatness-Based MPC

We propose using the deterministic perception constraint on output $\mathbf{y}$ in (13) in a flatness-based MPC framework. We will illustrate that in such a framework the only nonlinearity in the proposed OCP comes from this perception constraint. At each time step, we can solve our OCP by linearising (13) about the previously computed optimal trajectory.

To implement such a flatness-based MPC framework, we first show that our multirotor model described by (1a)-(1c), (2) and (3a)-(3b) is differentially flat in our output $\mathbf{y} = (x, y, z, \phi)$.

We recall the formal definition of differential flatness which requires us to show that both the state $\mathbf{x}$ and input $\mathbf{u}$ can be written as smooth functions of our output $\mathbf{y}$ and its derivatives.

Definition 1. A nonlinear system model $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$, $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{u} \in \mathbb{R}^m$, is *differentially flat* if there exists $\mathbf{y} \in \mathbb{R}^m$, whose components are differentially independent (that is, the components are not related to each other through a differential equation), such that the following conditions holds, see Fliess et al. (1995):

$$\text{Condition 1: } \mathbf{y} = \Lambda(\mathbf{x}, \mathbf{u}, \dot{\mathbf{u}}, \ldots, \mathbf{u}^{(\delta)}),$$
$$\text{Condition 2: } \mathbf{x} = \Phi(\mathbf{y}, \dot{\mathbf{y}}, \ldots, \mathbf{y}^{(\rho-1)}),$$
$$\text{Condition 3: } \mathbf{u} = \Psi^{-1}(\mathbf{y}, \dot{\mathbf{y}}, \ldots, \mathbf{y}^{(\rho)}),$$

where Λ, Φ and Ψ^{-1} are smooth functions, δ and ρ are the maximum orders of the derivatives of $\mathbf{u}$ and $\mathbf{y}$ needed to describe the system.

We illustrate that Conditions 1-3 hold for our multirotor model. We use the notation $c_\theta = \cos\theta$, $s_\theta = \sin\theta$ and $t_\theta = \tan\theta$. The rotation matrix elements are: $\mathbf{R}_{13} = c_\phi s_\theta c_\psi + s_\phi s_\psi$, $\mathbf{R}_{23} = c_\phi s_\theta s_\psi - s_\phi c_\psi$, $\mathbf{R}_{33} = c_\phi c_\theta$.
Condition 1: Condition 1 is true given that our output $\mathbf{y} = (x, y, z, \psi)$ comprises part of our state $\mathbf{x}$.
Condition 2: Given that for our multirotor state $\mathbf{x} = (x, y, z, \dot{x}, \dot{y}, \dot{z}, \ddot{z}, \phi, \theta, \psi)$, our output presents itself explicitly in some of the states, we only need to show that the roll ϕ and pitch θ can be written as smooth functions of the output and its derivatives. Using (3a)-(3b), we can write the pitch θ as a function of the derivatives of the output as:

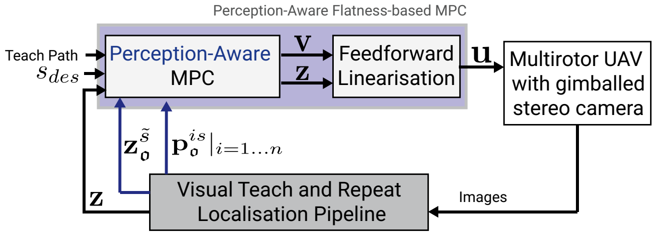

Perception-Aware Flatness-based MPC
Teach Path
s_{des}
Perception-Aware MPC
$\mathbf{v}$
$\mathbf{z}$
Feedforward Linearisation
$\mathbf{u}$
Multirotor UAV with gimballed stereo camera
$\mathbf{z}_{\mathfrak{o}}^{\tilde{s}}$
$\mathbf{p}_{\mathfrak{o}}^{is}|_{i=1...n}$
$\mathbf{z}$
Visual Teach and Repeat Localisation Pipeline
Images

Fig. 5. Block diagram of the proposed Perception-Aware Flatness-Based Model Predictive Control used during repeat to fly at some desired speed s_{des}. Including perception awareness into MPC requires additional inputs: 1) landmarks $\mathbf{p}_{\mathfrak{o}}^{is}$ from the trunk (closest vertex in the teach path) and 2) the current camera orientation $\mathbf{z}_{\mathfrak{o}}^{\tilde{s}}$. The Perception-Aware MPC involves solving the OCP (16) with constraint (13). To obtain the input $\mathbf{u}$, we use feedforward linearisation where the computed optimal input $\mathbf{v}$ and state $\mathbf{z}$ are fed through the inverse $\Psi^{-1}(\cdot)$ obtained from Condition 3.

$$t_\theta = c_\psi \frac{\mathbf{R}_{13}}{\mathbf{R}_{33}} + s_\psi \frac{\mathbf{R}_{23}}{\mathbf{R}_{33}} = \frac{c_\psi \ddot{x} + s_\psi \ddot{y}}{\ddot{z} + g}. \tag{14}$$

Similarly, we find the roll ϕ as:

$$t_\phi = (s_\psi \frac{\mathbf{R}_{13}}{\mathbf{R}_{33}} - c_\psi \frac{\mathbf{R}_{23}}{\mathbf{R}_{33}})c_\theta = \frac{s_\psi \ddot{x} - c_\psi \ddot{y}}{\ddot{z} + g} c_\theta, \tag{15}$$

where θ is found from (14).
Condition 3: From (1c), the yaw input ψ_{cmd} is simply written as the first derivative of our yaw ψ. Similarly, from (2), our z-velocity command $\dot{z}_{cmd}$ is written as a linear function of the derivatives of our z position up to the its third derivative. We, therefore, need to show that our pitch and roll commands θ_{cmd} and ϕ_{cmd} can be described by the derivatives of our output. To show this for our pitch command θ_{cmd}, we take the time derivative of (14) to find $\dot{\theta}$ which we use with the expression for θ in (1b). This gives us an expression for θ_{cmd} in terms of our output and its derivatives. A similar approach can be applied to obtain such an expression for ϕ_{cmd}.

As described in our work in Greeff et al. (2018), we can then use *feedback or feedforward linearisation* to rewrite our multirotor model described by (1a)-(1c), (2) and (3a)-(3b) as an equivalent linear model:

$$\dot{\mathbf{z}} = \mathbf{A}\mathbf{z} + \mathbf{B}\mathbf{v}$$

with state $\mathbf{z} = (x, \dot{x}, \ddot{x}, y, \dot{y}, \ddot{y}, z, \dot{z}, \ddot{z}, \psi)$ and input $\mathbf{v} = (\dddot{x}, \dddot{y}, \dddot{z}, \dot{\psi})$ where $(\dddot{x}, \dddot{y}, \dddot{z})$ are the positional jerks and $\dot{\psi}$ is the yaw rate. We consider a discretization of this linear model in our flatness-based MPC formulation to compute the optimal input $\mathbf{v}$. We convert the input $\mathbf{v}$ to the command $\mathbf{u}$ using feedback or feedforward linearisation found from Condition 3.

Our flatness-based MPC optimizes for input $\mathbf{v}$ by repeatedly solving the following OCP at each time step:

$$\min_{\mathbf{v}_{0\ldots N-1}} \sum_{k=1}^{N} (\mathbf{y}_k - \mathbf{y}_{ref,k})^T \tilde{\mathbf{Q}} (\mathbf{y}_k - \mathbf{y}_{ref,k}) + \sum_{k=0}^{N-1} \mathbf{v}_k^T \tilde{\mathbf{R}} \mathbf{v}_k$$
$$\text{subject to} \quad \mathbf{z}_{k+1} = \mathbf{A}\mathbf{z}_k + \mathbf{B}\mathbf{v}_k, \ \mathbf{y}_k = \mathbf{C}\mathbf{z}_k, \tag{16}$$

where N is the prediction horizon and both $\tilde{\mathbf{Q}}$ and $\tilde{\mathbf{R}}$ are positive definite matrices. This OCP reduces to solving a convex quadratic program (QP) at each time step. Our reference $\mathbf{y}_{ref}$ is the position (and associated yaw) along our teach path based on the desired speed s_{des}.

Introducing *perception awareness* into our MPC formulation then simply becomes solving the OCP (16) but with an additional constraint given by (13). The only nonlinearity in this formulation comes from the perception constraint (13). In standard real-time nonlinear MPC fashion, at each time step we linearize this constraint about the previously predicted optimal trajectory and perform one iteration of the SQP online. A block diagram of the full closed-loop architecture is shown in Fig. 5.

6. RESULTS

We fly the teach path shown in Fig. 6. However, we fly it with the camera pointed in different (but fixed) directions. Specifically, we first fly the teach path with the camera pointing toward the trees. This is Case 1, see Fig. 9 (a). We then fly the same teach path with the camera pointing toward the road. This is Case 2, see Fig. 9 (b). In both cases, we keep the camera at a constant downward pitch of 50° from the horizontal.

In each of these cases, we simulate the repeat path that would be flown using a flatness-based MPC (16) with (i) no path constraint, (ii) a fixed path constraint of 1 m, 2 m, 3 m, 4 m, 5 m and 6 m, and (iii) our proposed perception-aware (P-A) constraint governed by (13). In each of these cases and trials we run our outer-loop controller at 50 Hz. We implement our MPC with a discretization of 10 Hz and a 1.5 s prediction horizon (i.e. N = 15). We use the threshold for the minimum number of inliers $L_{min} = 30$. We consider our probabilistic inlier bound to be three standard derivations, i.e. $\bar{c} = 3$. In other words, we guarantee probabilistically that 99.9% of the time we will maintain more than 30 inliers.

Case 1 – Looking toward trees: As seen in Fig. 7(a), we cannot guarantee localisation when repeating the taught path using flatness-based MPC with no constraint (None) or with fixed path-error constraint of greater than or equal to 4 m. This is because there is some portion of our repeat path where our inlier bound $(\mu - \bar{c}\sigma)$ is less than our threshold of 30 inliers. Consequently, we cannot guarantee localisation (and thus complete the path when flying under vision).

While localisation is guaranteed when we simulate our repeat path using a flatness-based MPC with fixed path-error constraint of 3 m, we manage to achieve a *slightly* higher average speed (Fig. 7(b)) with our proposed perception-aware (P-A) constraint. As illustrated in Fig. 9(a), we are able to achieve a slightly higher average speed using our perception-aware constraint because we fly a profile with a path error between 3 m and 4 m. Consequently, our average speed when using our perception-aware constraint is between that flown under a fixed constraint of 3 m and 4 m in Fig. 7(b).

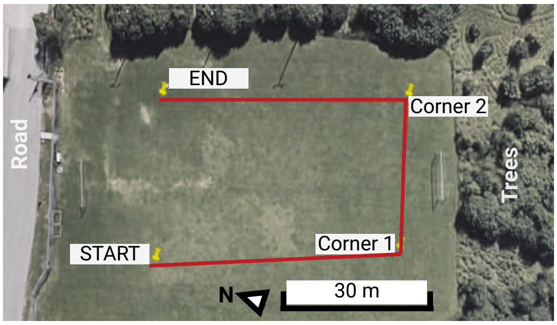


Fig. 6. Overview of the teach path (red) physically flown while looking toward the trees (Case 1) and toward the road (Case 2).

Moreover, a key benefit over using a fixed path error constraint is that we do not need to simulate (or fly) under different fixed path error constraints in order to find the constraint that ensures localisability. Instead, we explicitly guarantee it in our perception-aware constraint.

Case 2 – Looking toward road: As seen in Fig. 8(a), we cannot guarantee localisation when repeating the taught path using flatness-based MPC with no constraint (None) or with fixed path-error constraint of greater than or equal to 4 m.

While localisation is guaranteed when we simulate our repeat path using a flatness-based MPC with fixed path-error constraint of 3 m, we manage to achieve a *significantly* higher average speed (Fig. 8(b)) with our proposed perception-aware (P-A) constraint. In this case, our perception-aware approach achieves an average speed between that obtained with a fixed path error constraint of 4 m and 5 m. However, with a fixed path-error constraint of 4 m and 5 m we do not guarantee localisation (see Fig.8(a)). As illustrated in Fig. 9(b), under our perception-aware constraint, we fly a repeat profile that allows a path error greater than 4 m after the first corner, but then comes in to a 3 m path error after the second corner. This shows that the allowable error under which we can localise can significantly vary along the path. This is not only captured in our perception-aware constraint, but also exploited to achieve a higher overall speed.

Another key result highlighted in Fig. 9 is that under our perception-aware constraint, we do not repeat the taught path in the same way in Case 1 and Case 2. This is because it accounts for the landmark locations with respect to the path. In Case 1, the direction of our path error relative to the teach path after corner 1 means that we move closer to the scene. In Case 2, the direction of our path error relative to the teach path after corner 1 means that we move further from the scene. By moving further away from the scene we can actually afford a greater path error compared to when moving into the scene. This is because when moving away from the scene we tend to achieve a smaller parallax angle for the same path error and keep more landmarks within view. This effect is captured in our perception-aware constraint and is, consequently, why we fly the same geometric path differently just by looking in different directions.

7. CONCLUSION

We have presented an approach of building perception-awareness into flatness-based MPC which allows for optimal fast flight under reliable vision-based navigation. The key is that it does this by capturing:

- The variability of the localisation envelope (allowable path error under which we can still localise) along a path;
- The difference in the localisation envelope from one path to another. This includes geometrically identical paths that are flown at different altitudes, in different environments or with the camera pointed in different directions (as we illustrated in simulation based on real vision data).

Practically, the proposed perception-aware approach is also beneficial as it does not require 're-tuning' the allowable constraint for every new path or environment flown under vision-based navigation.

REFERENCES

Beul, M., Droeschel, D., Nieuwenhuisen, M., Quenzel, J., Houben, S.,& Behnke, S. (2018). Fast autonomous flight in warehouses for inventory applications. *IEEE Robotics and Automation Letters*, volume 3, pages 3121-3128.

Churchill, W., Tong, C. H., Gurau, C., Posner, I., & Newman, P., (2015). Know your limits: embedding localiser performance models in teach and repeat maps. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, pages 4238-4244.

DARPA (2015). Fast Lightweight Autonomy (FLA). solicitation number DARPA-BAA-15-16.

Falanga, D., Foehn, P., Lu, P., & Scaramuzza, D. (2018). PAMPC: Perception-aware model predictive control for quadrotors. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

Fliess, M., Lévine, J., Martin, P., & Rouchon, P. (1995). Flatness and defect of non-linear systems: introductory theory and examples. *International Journal of Control*, volume 61(6), pages 1327-1361.

Furgale, P. T. & Barfoot, T. D. (2010). Visual teach and repeat for long-range rover autonomy. In *Journal of Field Robotics*, volume 27(1), pages 534-560.

Gao, F., Wang, L., Wang, K., Wu, W., Zhou, B., Han, L., & Shen, S. (2019). Optimal trajectory generation for quadrotor teach-and-repeat. *IEEE Robotics and Automation Letters*, volume 4(2), pages 1493-1500.

Greeff, M., & Schoellig, A.P. (2018). Flatness-based model predictive control for quadrotor trajectory tracking. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6740-6745.

Gurau, C., Tong, C. H., & Posner, I. (2016). Fit for purpose? predicting perception performance based on past experience. In *Proc. International Symposium on Experimental Robotics (ISER)*, pages 454-464.

Gurau, C., Rao, D., Tong, C. H., & Posner, I. (2017). Learn from experience: probabilistic prediction of perception performance to avoid failure. *The International Journal of Robotics Research*, pages 1-15.

Kamel, M., Burri, M., & Siegwart, R. (2017). Linear vs nonlinear MPC for trajectory tracking applied to rotary wing micro aerial vehicles. *IFAC-PapersOnLine*, volume 50(1), pages 3463-3469.

Kaufmann, E., Gehrig, M., Foehn, P., Ranftl, R., Dosovitskiy, A., Koltun, V., & Scaramuzza, D. (2018). Beauty and the beast: optimal methods meet learning for drone racing. In *Proc. IEEE/RSJ International Conference on Robotics and Automation (ICRA)*, pages 690-696.

Mohta, K., Watterson, M., Mulgaonkar, Y., Liu, S., Qu, C., Makineni, A., Saulnier, K., Ke Sun, Zihao Zhu, A., Delmerico, J. A., Karydis, K., Atanasov, N., Loianno, G., Scaramuzza, D., Daniilidis, K., Jose Taylor, C., & Kumar, V. (2018). Fast, autonomous flight in GPS-denied and cluttered environments. *Journal of Field Robotics*, volume 35(1), pages 101-120.

Ostafew, C. J., Schoellig, A. P., Barfoot, T. D., & Collier, J. (2016). Learning-based nonlinear model predictive control to improve vision-based mobile robot path tracking. *Journal of Field Robotics*, volume 33(1), pages 133-152.

Patel, B., Warren, M., & Schoellig, A. P. (2019). Point me in the right direction: improving visual localization on UAVs with active gimballed camera pointing. In *Proc. Conference on Computer and Robot Vision (CRV)*, pages 105-112.

Warren, M., Greeff, M., Patel, B., Collier, J., Schoellig, A. P., & Barfoot, T. D. (2019). There's no place like home: visual teach and repeat for emergency return of multirotor UAVs during GPS failure. *IEEE Robotics and Automation Letters*, volume 4(1), pages 161-168.

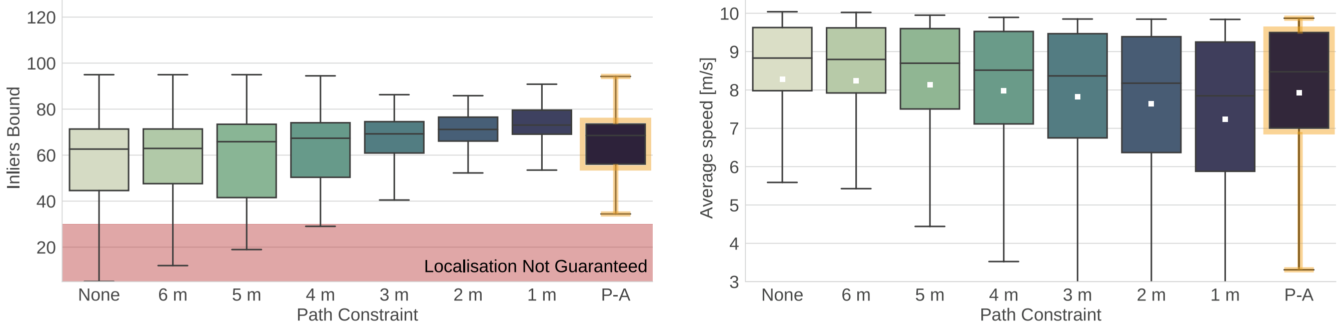
Appendix A. NORMAL DISTRIBUTION APPROXIMATION

We apply Lyapanov's Central Limit Theorem (CLT) to our Poisson binomial distribution in Sec. 5 to show that it converges to a normal distribution in its limit.

Theorem 1. (Lyapanov's CLT). Suppose $\{X_1, X_2, ..., X_n\}$ are independent (not necessarily identically distributed) random variables each with a finite mean μ_i and variance σ_i^2. If for some $\rho > 0$, $\lim_{n\to\infty} \frac{1}{s_n^{\rho+2}} \sum_{i=1}^{n} \mathrm{E}[|X_i - \mu_i|^{\rho+2}] = 0$, where $s_n^2 = \sum_{i=1}^{n} \sigma_i^2$, then the Central Limit Theorem holds, i.e. $\frac{1}{s_n} \sum (X_i - \mu_i) \to \mathcal{N}(0, 1)$ as $n \to \infty$.

In Sec. 5 we treat the landmarks as independent Bernoulli random variables $l_i \sim \mathrm{Ber}(p_i)$ where the first two moments of each variable, $\mu_i = p_i$ and $\sigma_i^2 = p_i(1-p_i)$, are finite. We can show that Theorem 1 holds by, for example, selecting $\rho = 2$. We use the fact that for a Bernoulli random variable $\mathrm{E}[X_i^k] = \mathrm{E}[X_i] = p_i, \forall k$, to show:

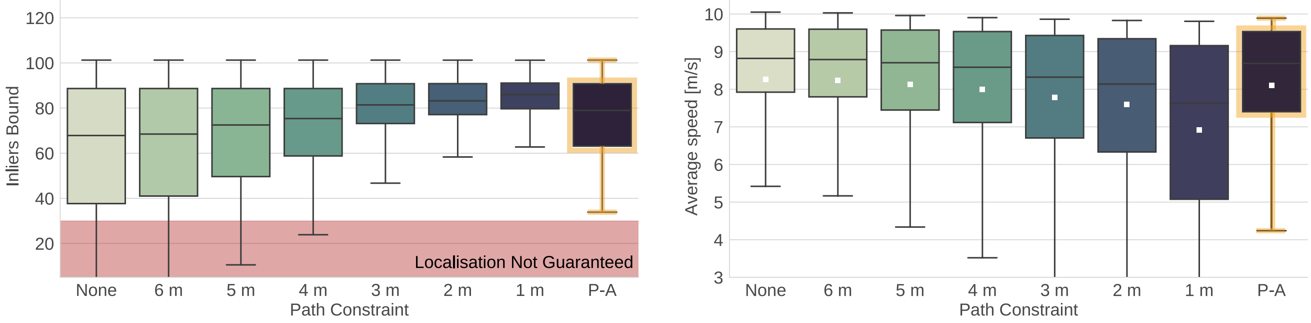
$\mathrm{E}[|X_i - p_i|^4] = p_i(1-p_i) - 3p_i^2(p_i-1)^2 \leq p_i(1-p_i) = \sigma_i^2$. Therefore, $\frac{1}{s_n^4} \sum_{i=1}^{n} \mathrm{E}[|X_i - p_i|^4] \leq \frac{1}{s_n^2}$. We are now left to show that $\frac{1}{s_n^2} \to 0$ as $n \to \infty$. For $0 < p_i < 1$, $s_n^2 = \sum_{i=1}^{n} \sigma_i^2 = \sum_{i=1}^{n} p_i(1-p_i) \to \infty$ as $n \to \infty$. Therefore, $\frac{1}{s_n^2} \to 0$ as $n \to \infty$. This shows that Lyapanov's CLT holds which justifies the normal approximation made in Sec. 5.



(a) Inlier Bound

(b) Average Speed

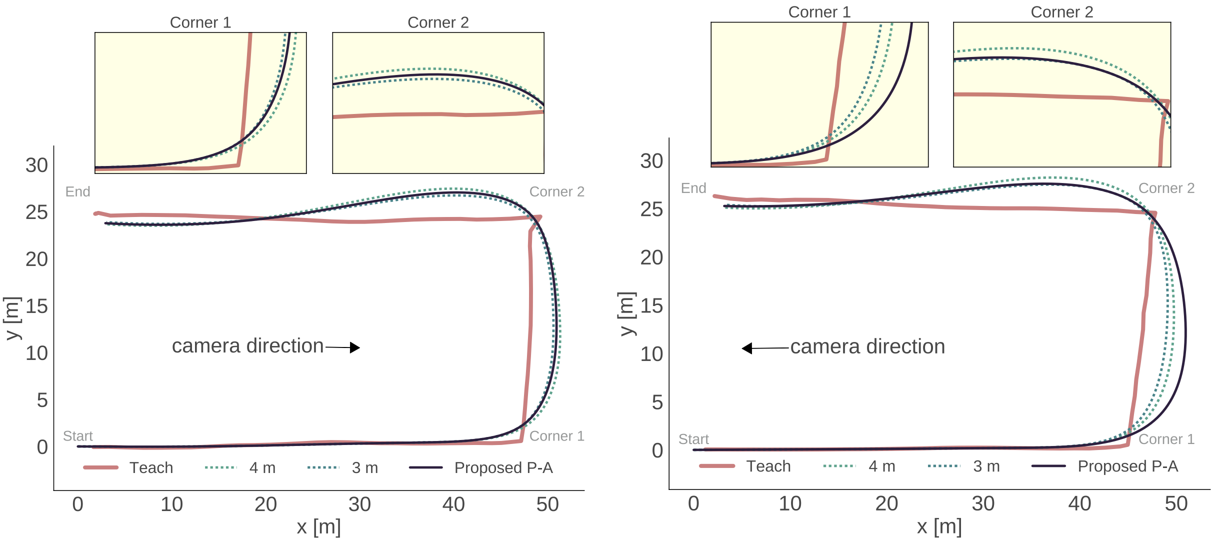
Fig. 7. *Case 1 – Looking toward trees:* (a) inlier bound for guaranteed localisation ($\mu - \bar{c}\sigma$) and (b) average speed (m/s) when repeating the teach path (taught facing the camera toward the trees) with a desired speed $s_{des} = 10$ m/s using a flatness-based MPC with (i) no constraint (None), (ii) a fixed path error constraint (1 m, 2 m, 3 m, 4 m, 5 m, 6 m), and (iii) our proposed perception-aware (P-A) constraint given by (13).



(a) Inlier Bound

(b) Average Speed

Fig. 8. *Case 2 – Looking toward road:* (a) inlier bound for guaranteed localisation ($\mu - \bar{c}\sigma$) and (b) average speed (m/s) when repeating the teach path (taught facing the camera toward the road) with a desired speed $s_{des} = 10$ m/s using a flatness-based MPC with (i) no constraint (None), (ii) a fixed path error constraint (1 m, 2 m, 3 m, 4 m, 5 m, 6 m), and (iii) our proposed perception-aware (P-A) constraint given by (13).



(a) Case 1 – Looking toward trees

(b) Case 2 – Looking toward road

Fig. 9. (a) Simulated path profile of repeating taught path in Case 1 using flatness-based MPC with (i) a fixed path error constraint of 3 m, (ii) a fixed path error constraint of 4m, (iii) our perception-aware constraint (P-A); (b) Simulated path profile of repeating taught path in Case 2 using flatness-based MPC with (i) a fixed path error constraint of 3 m, (ii) a fixed path error constraint of 4 m, (iii) our perception-aware constraint (P-A).