

Thermal Image Based Navigation System for Skid-Steering Mobile Robots in Sugarcane Crops*

Marco F. S. Xaud¹, Antonio C. Leite² and Pål J. From¹

Abstract—This work proposes a new strategy for autonomous navigation of mobile robots in sugarcane plantations based on thermal imaging. Unlike ordinary agricultural fields, sugarcane farms are generally vast and accommodates numerous arrangements of row crop tunnels, which are very tall, dense and hard-to-access. Moreover, sugarcane crops lie in harsh regions, which hinder the logistics for employing staff and heavy machinery for mapping, monitoring, and sampling. One solution for this problem is TIBA (Tankette for Intelligent BioEnergy Agriculture), a low-cost skid-steering mobile robot capable of infiltrating the crop tunnels with several sensing/sampling systems. The project concept is to reduce the product cost for making the deployment of a robot swarm feasible over a larger area. A prototype was built and tested in a bioenergy farm in order to improve the understanding of the environment and bring about the challenges for the next development steps. The major problem is the navigation through the crop tunnels, since most of the developed systems are suitable for open field operations and employ laser scanners and/or GPS/IMU, which in general are expensive technologies. In this context, we propose a low-cost solution based on infrared (IR) thermal imaging. IR cameras are simple and inexpensive devices, which do not pose risks to the user health, unlike laser-based sensors. This idea was highly motivated by the data collected in the field, which have shown a significant temperature difference between the ground and the crop. From the image analysis, it is possible to clearly visualize a distinguishable corridor and, consequently, generate a straight path for the robot to follow by using computationally efficient approaches. A rigorous analysis of the collected thermal data, numerical simulations and preliminary experiments in the real environment were included to illustrate the efficiency and feasibility of the proposed navigation methodology.

I. INTRODUCTION

One of the major challenges encountered in agricultural robotics is to deal with the autonomous navigation in different environment types [1]. Although several solutions for this problem are fairly established for small open fields and structured indoor facilities [2], [3], such as greenhouses and nurseries, the issue remains unsolved for grass-type crops, such as sugarcane, wheat, corn and barley, which are large, unstructured and dense environments. In addition, most of the existing solutions employ laser scanners and GPS/IMU for an accurate robot mapping, which is considerably expensive, especially if the robot has processing

and cost limitations for the electromechanical design. One example of robotic vehicle for grass-type crops is TIBA (Tankette for Intelligent BioEnergy Agriculture), a skid-steering mobile robot (SSMR) intended to carry out a number of agricultural tasks in sugarcane fields [4]. In the current development and experimental phase at the UMOE BioEnergy fields (Fig. 1a), in Presidente Prudente, São Paulo state, Brazil, TIBA is being tested for mapping of general plant health using NDVI, detection and control of pests, weeding, and stalk/soil sampling. SSMRs are suitable for all-terrain purposes due to their robust mechanical structure able to work in hard environmental conditions [5]. However, for small velocities and low accelerations, SSMRs behave approximately as nonholonomic systems, which impose great challenges for the implementation of the control system. Moreover, the navigation problem for this robot is especially involved because of: (i) the need for driving through tight crop corridors with uneven terrain, trapping leaves, poor GPS coverage and hard to interpret data from RGB camera or lasers; (ii) lack of maturity for skid effects analysis due to the required slippage phenomenon for this driving profile, despite the existence of relevant research on motion control and path planning for nonholonomic systems [5], [6]; (iii) limited hardware and processing capacity, due to its low-cost design concept with motivation for deployment of swarm-based approach [4].

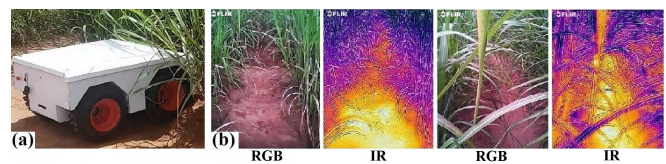


Fig. 1. TIBA Robot during field tests - (a) crop tunnel infiltration; (b) RGB and IR images from same place in the tunnel. It is possible to see further beyond using IR, especially in cloudy/dark daytime or in cases in which the view is very occluded by leaves.

During the field experiments, several thermal IR media were collected from the crop tunnels by using a FLIR[®] One V2 camera [4], which was motivated by the ability of IR imaging to help identifying objects under limited visibility conditions with minimal attenuation, such as night, light dust, smoke, and haze [7]. Notice in Fig.1b that, when compared to regular RGB pictures, the IR images provide a much more distinguishable shape of the crop corridor ground along which the robot must navigate. The technology for passive IR imaging has recently become cheap and portable to be mounted on small robotic platforms. It detects thermal radiation naturally emitted from objects

*This work was partially supported by the UTFORSK Partnership Programme from The Norwegian Centre for International Cooperation in Education (SIU), project number UTF-2016-long-term/10097.

¹Marco F. S. Xaud and Pål J. From are with the Norwegian University of Life Sciences, Faculty of Science and Technology, P. O. Box 5003 - 1432, Ås, Norway. E-mail: [mafernan, pal.johan.from]@nmbu.no

²A. C. Leite is with the Pontifical Catholic University of Rio de Janeiro, Department of Electrical Engineering, Postal code 22451-900, Rio de Janeiro RJ, Brazil. E-mail: antonio@ele.puc-rio.br

and, therefore, dispenses active sources for transmission of thermal energy to a target, which mitigates life risks offered by laser-based mapping. Given this background and the context of very slow speeds, there is enough room for testing state-of-the-art image segmentation approaches with long detection delays that provide a reliable reference trajectory to be followed, such as artificial intelligence, especially deep convolutional neural networks (CNN) [8]. In agriculture, we can highlight the use of fully convolutional networks (FCNs) for weed classification for plant-specific treatment [9]. Combination of the CNN approaches with IR imaging were already addressed, e.g., for rail detection [10], health monitoring [11], wildlife recognition [12], and urban traffic scene segmentation [13], but not for the navigation problem in agricultural fields, according to the best of author's knowledge. Verification and validation are carried out through the inference of the generated path from thermal images collected in the field, and from numerical simulations¹. Preliminary experimental tests performed in sugarcane crops also illustrate the efficiency and feasibility of the proposed navigation methodology, which is composed of different implementation phases (Fig. 2).



Fig. 2. Problem formulation: different implementation phases.

II. MODELING OF ROBOT AND ENVIRONMENT

In this work, we consider that the working environment is of the same type as that encountered in the sugarcane field where the robot was first tested, which is a set of crop rows with seeding points predefined by GPS coordinates. Those points lie over circles of radius approximately between 30 and 40 cm, with distance of 1.50 m between their centers, which provides a gap corridor where the robot should be driven (Fig. 3a). For modeling purposes, we simply consider the environment as a locally flat surface and the kinematic model of the SSMR, according to the approach proposed by [5]. First, let us define two reference frames as shown in Figs. 3b-c: the global or the *inertial frame* $\bar{E}_0 = \{ \bar{x}_0, \bar{y}_0, \bar{z}_0 \}$ and the local or the *robot frame* $\bar{E}_r = \{ \bar{x}_r, \bar{y}_r, \bar{z}_r \}$. The origin of \bar{E}_r is assigned to the robot center-of-mass (CM), \bar{x}_r points towards the vehicle front (or sagittal axis), and \bar{z}_r is normal to the robot chassis. The distances a , b , and c defines the position of the robot CM with respect to the wheels' centers. The SSMR is a vehicle with four single-orientable wheels and its configuration is completely described by the state vector $q = [X Y \phi]^T$, which denotes the robot CM generalized coordinates, where the pair (X, Y) are the Cartesian coordinates of the CM and ϕ is the orientation angle (or angular difference) of the robot frame \bar{E}_r with respect to the inertial frame \bar{E}_0 . Notice that, in this context, the state vector $\dot{q} = [\dot{X} \dot{Y} \dot{\phi}]^T$, denotes the generalized velocities of the robot. From the literature, it is well know

¹More illustrations of the navigation methodology can be viewed in the accompanying video clip at <https://youtu.be/unnbk5y31s>.

that SSMR have nonholonomic (or non-integrable) kinematic constraints [14], which introduces extra challenges for modeling and control design. Moreover, the slippage phenomenon (mandatory for the SSMR driving profile [5]) determines the position of the robot *instant center of rotation* (ICR), which is not unique. The ICR depends on the desired drift, which may lead to system instability if these quantities are not well modeled. From the kinematic model introduced in [5], it is possible to obtain the relationship between the SSMR velocities and its wheel angular speeds as $\eta = [v_x \ \omega]^T = r [(\omega_L + \omega_R)/2 \ (-\omega_L + \omega_R)/(2c)]^T$, where $v_x, \omega \in \mathbb{R}$ are respectively the robot driving and steering velocities, $r > 0$ is the wheel radius, and $\omega_L, \omega_R \in \mathbb{R}$ are the angular velocity of the left and right wheels, respectively. The motion of the SSMR is described by the differential kinematic equation as $\dot{q} = S(q) \eta = [s_1(q) \ s_2(q)] \eta$ where $s_1(q) = [\cos(\phi) \ \sin(\phi) \ 0]^T$ and $s_2(q) = [x_{icr} \sin(\phi) \ -x_{icr} \cos(\phi) \ 1]^T$ are input vector fields, with x_{icr} being the robot ICR position along \bar{x}_r . The kinematic model also has the non-integrable speed constraint $v_y + x_{icr} \dot{\phi} = 0$, which is written in the Pfaffian form as $A(q) \dot{q} = 0$ with $A(q) = [-\sin(\phi) \ \cos(\phi) \ x_{icr}]$, showing that, in the absence of slipping, the contact point speed is zero in the direction orthogonal to \bar{x}_r [14].

III. ROBOT LOCALIZATION

In this phase, our goal is to obtain the robot localization with respect to the crop row centerline. Since we seek for a simple and inexpensive strategy to ensure that the robot follows a straight line, it is not necessary to find its position along the x_0 -axis of the corridor path. This phase is divided in two steps: (i) *Runway centerline estimation*, which comprises all the procedures related to the segmentation processing of thermal image/video to extract the almost triangular shaped object of the corridor ground (which resembles a runway in perspective) and the centerline that points towards the path vanishing point; (ii) *Offset generation*, in which we estimate and quantify the lateral displacement e_y and misalignment angle e_ϕ of the robot sagittal axis from the row centerline by using the information obtained in the previous step and the robot/camera parameters. This phase is arguably the most relevant in the context of this research, since it opens a wide room for comparing the efficiency of different strategies to be investigated for each step and tested under several conditions, e.g., image noise, partial occlusion of camera lens by leaves, uneven terrains causing robot rotation around x, y - axes, and round off error. Here, we present two different approaches to detect the runway feature, and a straightforward calculation of the robot offset.

A. Color Thresholding with Linear Regression (CT-LR)

In this section, we describe the image segmentation methodology used for runway centerline estimation. In this context, our first goal is to explicitly find the “hot” areas of the thermal picture, which is expected to match the ground. From the temperature color scale used in the collected images from the field, the hot temperatures lie within a color

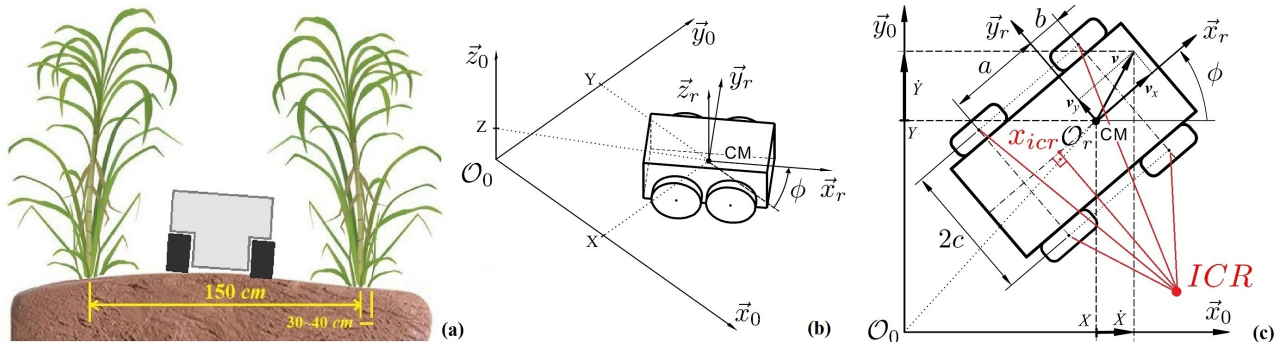


Fig. 3. Robot and environment modeling: (a) crop corridor; (b) robot coordinates; (c) dimensions and center-of-mass. Images adapted from [5].

spectrum around yellow, orange, red and white. An empirical analysis of all the collected media has shown that a simple color threshold (CT) technique is capable of solving most of the cases. In this work, the CT technique was set to a hue range within $\mathcal{H} \in [0.145, 0.944]$, by using the Hue-Saturation-Value (HSV) color representation (Figs. 4a-b).

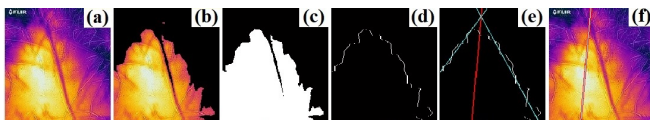


Fig. 4. Color thresholding (CT) application example with Linear Regression (LR): (a) original frame; (b) CT; (c) binary image shape; smaller pixel groups removed; (d) extract shape contour by selecting only the white pixels adjacent with background; (e) apply independent LR for “left” and “right” pixel groups (fitted cyan lines), then compute their average (red); (f) result.

Then, after the application of the CT technique, several approaches can be used to estimate a straight line as the center of the triangular feature of the “hot ground runway” (Fig. 4c, after removal of smaller pixel groups). One example is described in Fig. 4d-f, which final result shows that the fitted runway centerline is apparently very satisfactory for the navigation purpose.

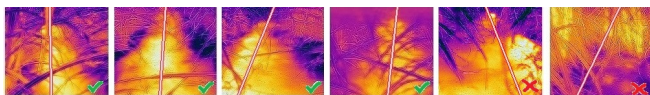


Fig. 5. Color thresholding (CT) application over several examples: green check for successful results and red cross for fails. For those real pictures, the criteria used to determine the success of the line fitting was partially subjective, but for all of them, we checked if the line crossed the runway peak and if it is kept within the runway area.

Figure 5 shows the result of applying CT for several example frames obtained from the sugarcane crop. Notice that this approach fails for few cases. We can highlight the 6-th frame (Fig. 5), which is a crop condition that presents “tall leaves” or “colder terrains”. In this case, less sunlight reaches the ground, making it apparently colder than the plants, and hence, requiring different color thresholds to be tuned.

B. Image Semantic Segmentation Using Deep Learning

In the last decade, the *Deep Learning* framework has been widely used for numerous applications in many fields of research and engineering, with strong appeal for computer vision in robotics [8]. Supervised learning using convolutional

neural networks (CNNs) for image semantic segmentation of images has enormous potential to overcome situations in which the application of a simple CT approach provides unsatisfactory results [15], [16], [17], [18]. This is especially true in the recognition of occluded objects, as described by [19], [20]. A high computational cost is expected for the training step, but not for the inferring process during the robot field operation, notably for navigation at low speeds. In our case, we want to train a CNN which is able to delimit a polygon over the “hot runway” [21], even if this feature is under undesired conditions or has unusual shape (Fig. 6).

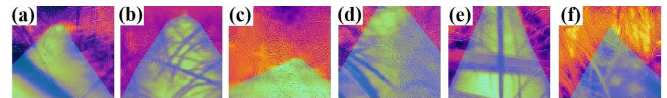


Fig. 6. Bad situations and expected features - (a) erratic curved shape; (b) blurry; (c) small color/temperature difference; (d) noise/many branches; (e) leaves trapping occluding image; (f) colder environment/tall plants.

Due to the great amount of collected images in the field, simple CNNs were trained with OpenCV library to test a preliminary segmentation of the given media and to compare the obtained results with the feature extracted via CT. A set of 89 downsampled examples were selected from the dataset to be the input training set, and the respective outputs were defined as binary images representing two different classes: ‘background’ for black and ‘runway’ for white. Different kernel widths, strides, pooling and padding values were tested on a 10-layer fully convolutional neural network (FCN). In future works, a more sophisticated evaluation of this problem will be performed in order to investigate strategies that best use CNNs to cope with the situations commonly found in the crop tunnels. The obtained results point out satisfactory contributions for the “runway” feature detection. Figure 7 shows the output results for some examples validated from the trained CNNs. Notice in those two specific cases (Fig. 7a) that the CNN approach partially overcomes the problem of leaf occlusion (Fig. 7b), whereas CT technique breaks up the feature into smaller regions (Fig. 7c). A mini-batch accuracy of 89.79% was obtained after 100 epochs. For the early comparison of the required computational costs between CNN and CT algorithms, a set of 180 images were selected for validation by using both methods. The mean and standard deviation of the elapsed

processing time (in ms) were $(\mu_{ct}, \sigma_{ct}) = (0.37, 0.16)$ for the CT method and $(\mu_{cnn}, \sigma_{cnn}) = (815.90, 175.80)$ for the CNN, which is much longer, but still small enough for the expected speed and detection rate of this navigation problem. From a careful analysis, it is straightforward to conclude that the combination of the CNN results and the linear regression approaches explained in Section III-A enhances the estimation of the corridor centerline, as shown in Fig. 8.

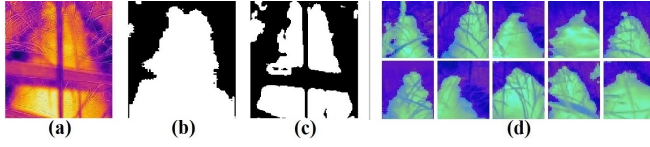


Fig. 7. Example from trained CNN: (a) original input; (b) runway extracted by CNN; (c) same by CT; (d) some CNN results displayed by output overlay over input. For this test, a PC with Intel® Core™ i7-7700HQ CPU @ 2.80GHz, 16GB RAM and nVIDIA GeForce™ GTX-1050 4GB GDDR5 GPU was used. The 10-layer CNN was composed by the following layers: Image Input, Conv. 2D, ReLU, Max. Pooling 2D, Conv. 2D, ReLU, Transp. Conv. 2D, Conv. 2D, Softmax, and Output Pixel Classification.

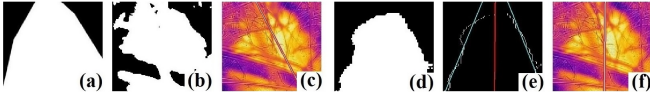


Fig. 8. CNN and linear regression - (a) expected feature; (b) runway from CT; (c) centerline estimation from CT-LR; (d) runway from CNN; (e) linear regression over CNN; (f) centerline estimation from CNN-LR.

The use of CNNs evidences even more the benefits of using thermal images instead of conventional RGB ones. This is revealed in Fig. 9, which displays some validation results from two differently trained CNNs for each input image type, being both CNNs set with the same parameters.

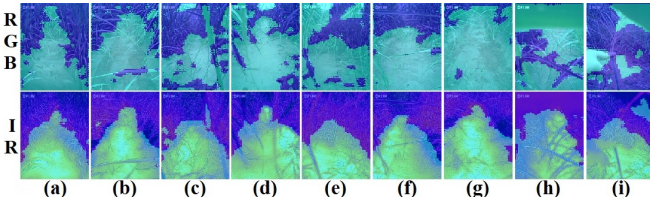


Fig. 9. CNN for thermal IR images with mini-batch accuracy of 88.95%, 150 epochs, and for RGB images with mini-batch accuracy of 70.56%, 150 epochs. Some results with inferred features highlighted - (a-b) CNN-RGB with acceptable output, but worse than CNN-IR; (c-e) CNN-RGB fails and provide noisy output; (f-g) CNN-RGB fails and provide one big feature group; (h-i) occluding leaf at top and intruding human hand at left, both ignored by CNN-IR, but misleading CNN-RGB.

However, the segmentation of the “cold ground runway” case has shown to be still unsatisfactory when we are dealing with ‘hot’ and ‘cold’ inputs at the same time, which therefore demands a more complex learning network. Nevertheless, in this work, we investigate a topology that combines three CNNs. The first one (CNN1) should classify the image between ‘hot’ (or short plants) and ‘cold’ (or tall plants). The result activates either a second or a third segmentation CNN, which are parallel and independent. The difference between CNN2 and CNN3 is that the third one is trained only with examples containing the “cold” case. Fig. 10a shows satisfactory results related to some test image inputs

of “cold” features to a trained CNN3. Notice the remarkable difference existing from a simple color thresholding application, which provides a noisy and abnormal shape. Finally, a 7-layer CNN1 was trained to classify the input between ‘hot’ and ‘cold’, which also presented a satisfactory behavior (Figs. 10b-c).

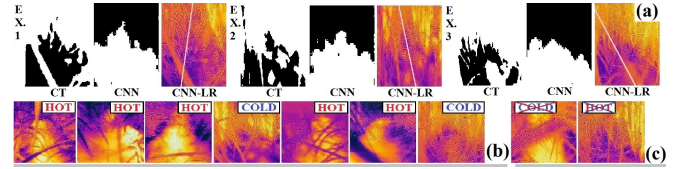


Fig. 10. Treatment of “cold” runway features - (a) results from CNN3; (b) results from trained CNN1 for classification, with 47 examples from test set and 90 from validation set (80 correctly classified inputs, or 94.4%); (c) examples of classification failures from CNN1 (5 frames or 5.6%).

C. Estimation of the Robot Offset from Corridor Centerline

The goal of this phase is to estimate the robot offset (e_y, e_ϕ) from the corridor ground centerline, as defined in early Section III. For better description, consider the image projection problem and the frames depicted in Fig. 11, where the coordinates of a target object, denoted by an arbitrary point P_i lying on the ground, is expressed in the camera frame $\bar{E}_c = \{\bar{x}_c, \bar{y}_c, \bar{z}_c\}$ by a position vector $r_i = [x_i \ y_i \ z_i]^T$ connecting its origin \mathcal{O}_c to that point. Notice that, $r_i \in \mathbb{R}^3$ always crosses the inner limits of the *sensor plane*, which consists of a rectangular array of photosensitive elements, termed photosites or pixels, with dimensions of α_x (width) and α_y (height) measured in millimeters.

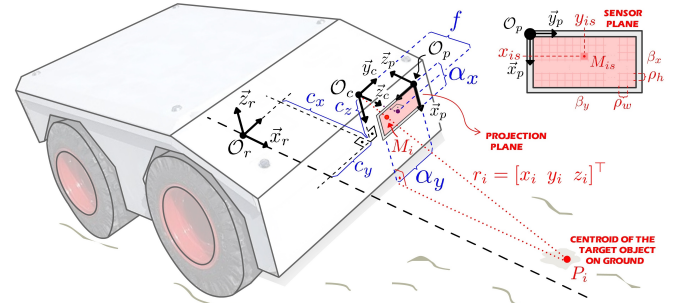


Fig. 11. Camera model and relevant frames: frontal perspective transformation. Here, without loss of generality, the centroid of the target object can be chosen as the arbitrary point P_i .

This array can be interpreted as a virtual plane, being the dimension of the image projected on it limited by the camera lens aperture. The projected image of an arbitrary point P_i , denoted by the point M_i , can be expressed in the *projection frame* $\bar{E}_p = \{\bar{x}_p, \bar{y}_p, \bar{z}_p\}$ by the 3-dimensional vector $M_{ip} = [x_{ip} \ y_{ip} \ 0]^T = (f/z_i)[x_i \ -y_i \ 0]^T$ or in the *sensor frame* $\bar{E}_s = \{\bar{x}_s, \bar{y}_s, \bar{z}_s\}$ by the 2-dimensional vector $M_{is} = (x_{is}, y_{is})$, where $x_{is}, y_{is} \in \mathbb{Z}_+^*$ are the image coordinates expressed in pixel units [22], and $f > 0$ is the camera lens focal length. The origin of both projection and sensor frames (\mathcal{O}_p and \mathcal{O}_s , respectively) are located at the upper left corner of the sensor plane. If we consider the sensor pixel resolution of β_x (rows) \times β_y (columns), the

dimension of a single pixel is then defined by $\rho_h = \alpha_x/\beta_x$ and $\rho_w = \alpha_y/\beta_y$, with the centroid of the top-left hand pixel located at the coordinates $x_{ip} = \rho_h/2$ and $y_{ip} = \rho_w/2$. Then, we can define the relation between the point M_i represented by pixel and its respective position referred in \bar{E}_p as $M_{ip} = [\rho_h x_{is} - \rho_h/2 \quad \rho_w y_{is} - \rho_w/2 \quad 0]^\top$. In general the camera frame \bar{E}_c is aligned with the center of the sensor plane and it is located at the coordinates (c_x, c_y, c_z) expressed in the robot frame \bar{E}_r . The image projection problem consists of finding the coordinates of a set of arbitrary points P_i for $i = 1, 2, \dots, n$ from a given input image, generated in the previous step (Subsection III-A). Such an image provides the result of the LR method applied over the original IR image, from which we extract a set of points M_{is} . Since the arbitrary points P_i are located at the target objects on the corridor ground centerline and expressed with respect to the \bar{E}_r , these are denoted by P_{ir} . Once the point M_{ip} is obtained, it is necessary to compute the object image projection M_{ir} referred to the robot frame \bar{E}_r , which is given by $\tilde{M}_{ir} = T_{rc} T_{cp} \tilde{M}_{ip}$, being $\tilde{M}_{ip} = [M_{ip}^\top \quad 1]^\top$, $\tilde{M}_{ir} = [M_{ir}^\top \quad 1]^\top \in \mathbb{R}^4$. Notice that T_{rc} and T_{cp} are homogeneous transformations matrices [14], which are given by:

$$T_{rc} = \begin{bmatrix} R_y(\frac{\pi}{2}) \bar{R} & (\vec{p}_{rc})_r \\ 0_{1 \times 3} & 1 \end{bmatrix}, T_{cp} = \begin{bmatrix} R_x^\top(\pi) & (\vec{p}_{cp})_c \\ 0_{1 \times 3} & 1 \end{bmatrix}, \quad (1)$$

where $(\vec{p}_{cp})_c = \frac{1}{2} [-\alpha_x \quad \alpha_y \quad 2f]^\top$, $(\vec{p}_{rc})_r = [c_x \quad c_y \quad c_z]^\top$, $R_x, R_y \in \mathbb{SO}(3)$ are elementary rotations around the x - and y -axis, respectively, and $\bar{R} \in \mathbb{SO}(3)$ is a composite rotation matrix to be used if \bar{E}_c and \bar{E}_r are not aligned.

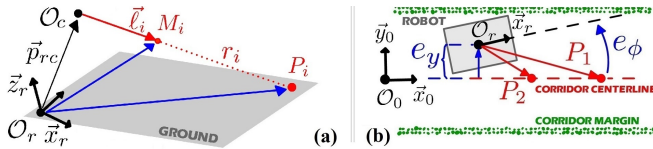


Fig. 12. Geometry: (a) object position; (b) corridor centerline offset.

To compute P_{ir} , let us first define the vector \vec{l}_i that connects the camera frame origin O_c to the point M_i (Fig. 12a) and is expressed in \bar{E}_r as $(\vec{l})_{ir} = M_{ir} - (\vec{p}_{rc})_r$. If we extend \vec{l}_i straightly along the line r_i by a multiplier factor $\gamma_i \in \mathbb{R}$, we obtain $P_{ir} = (\vec{p}_{rc})_r + \gamma_i(\vec{l})_{ir}$, which, considering both the robot base and ground aligned, yields $\gamma_i = -(z^\top M_{ir})/(z^\top (\vec{l})_{ir})$. Hereafter, we define x, y, z as orthogonal elementary unit vectors. Thereby, it is finally possible to find the robot offset $e = [e_y \quad e_\phi]^\top$ from the corridor centerline (Fig. 12b) with respect to an inertial frame \bar{E}_0 whose x_0 -axis points forward with respect to the robot motion direction. In this work, we consider two arbitrary points P_{1r}, P_{2r} previously obtained from M_{1s}, M_{2s} , respectively (with both points obligatory belonging to the image runway triangle border and to the fitted straight line via LR) and the vector $v_p = P_{1r} - P_{2r}$. Notice from Fig. 12b that P_{1r} must be assigned ahead of P_{2r} . The offset e_y that defines the minimal distance from O_r to the lane centerline is the norm of $v_\perp = (v_p v_p^\top) P_{1r} / \|v_p\|^2 - P_{1r}$, which is orthogonal to P_{1r}

projection onto the centerline, that is $e_y = \text{sgn}(y^\top v_\perp) \|v_\perp\|$. Then, e_ϕ is given by $e_\phi = \text{sgn}(z^\top \omega_p) \cos^{-1}(x^\top v_p / \|v_p\|)$, where $\text{sgn}(\cdot)$ denotes the sign function and $\omega_p = v_p \times x$ is a cross product operation.

Remark 1: It is worth mentioning that, for the sake of space saving and better readability, the trajectory planning algorithm and the motion control strategy for the skid-steering mobile robot (SSMR), depicted in the block diagram of Fig. 2, were both omitted here.

IV. VERIFICATION AND VALIDATION

To verify and validate the proposed navigation methodology, a 3D simulation environment was developed using MATLAB/Simulink. In addition, a 3D model of the sugarcane corridor and the mobile robot was created, so that, before the implementation in the real prototype, we could perform exhaustive tests of the combined navigation phases, considering different experimental setups and control strategies. More illustrations of the 3D simulation environment can be found in the accompanying video clip¹.

A. 3D Crop Environment and Virtual Thermal Map

The 3D crop environment (Fig. 13a) is a 15 m corridor of cylindrical stems equally spaced along the lane by 40 cm and other randomly distributed stems around each of them. The row and plant spacings are equal to the distances commonly found in sugarcane farms (Figs. 13b-e).

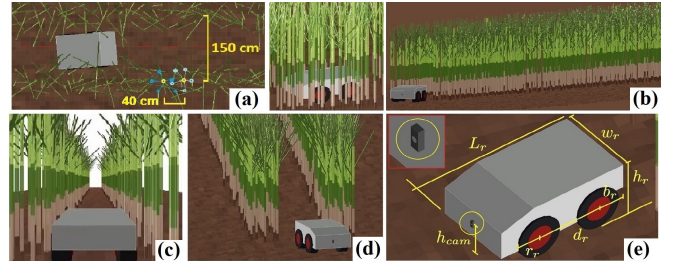


Fig. 13. 3D environment - (a) corridor top view, stems placed along 15 m, with central stems (yellow circles) separated by 40 cm and 5 stems (blue circles) around each of them, with random distance to the central stem with 25 cm of mean and 2 cm of standard deviation; (b-d) perspectives of robot and corridor entrance; (e) robot and camera 3D model, main dimensions (in cm): $L_r = 120$, $d_r = 58$, $r_r = 20$, $b_r = 26$, $h_r = 52.7$, $w_r = 72$, $h_{cam} = c_z = 22$, and $\bar{R} = R_y(15^\circ)$ (small camera fixed pitch).

The robot and IR camera models are designed to match the existing robot prototype and a real thermal imaging camera. A virtual thermal map (Fig. 14a) was created from the thermal images of the 3D sugarcane corridor. To achieve this, the ground was divided into a 5 cm cell-grid with a starting temperature of 70°C assigned to each cell. The position of each plant stem defined a leaf pattern (Fig. 14b) that subtracted a specific amount of temperature from the neighbour cells as a shadow effect. A color was assigned to each cell based on the color scale used by the IR camera, and this procedure was carried out empirically based on the thermal images collected in the field. To verify and validate the centerline estimation approach, the robot/camera objects were set to different poses along the corridor (Fig. 15). Notice

that, for different configurations, the maximum error between the true and the estimated robot pose offset was no more than $e_y \approx 4.1 \text{ cm}$ and $e_\phi \approx 1.9^\circ$, which is a satisfactory result for collision-free navigation along at least 5 m , until a new path is generated.

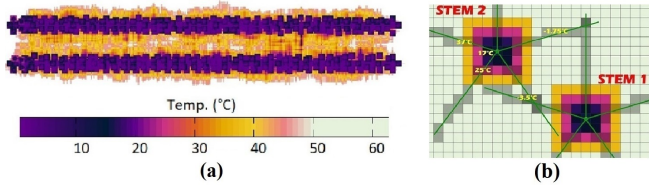


Fig. 14. Virtual thermal map - (a) corridor with temperature color scale; (b) temperature decrements by shadows from leaves placed between 72° , with smaller temperatures closer to the stalk core and under leaf crossings (in this figure, the stalks are more distant from each other for better illustration).

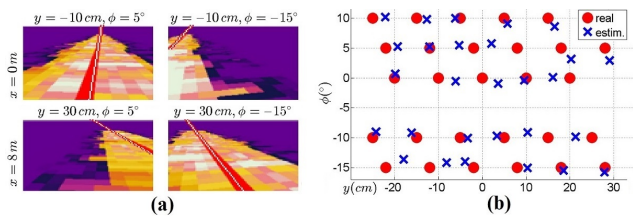


Fig. 15. Batch estimation - (a) different poses, true centerline in thick red, estimation in thin white; (b) comparison between real and estimated offsets. The virtual camera has a resolution of 120×160 pixels with intrinsic parameters set to $\rho = \rho_h = \rho_w = 12 \mu\text{m}$, $\alpha_x = 1440 \mu\text{m}$ and $\alpha_y = 1920 \mu\text{m}$, according to the datasheet of FLIR[®] One V2, which was the model actually used to collect the field images. The camera orientation was set to landscape so as to cover more horizontal field-of-view.

B. Navigation in Computing Environment

To demonstrate the effectiveness of the proposed navigation methodology, a trajectory was defined based on the references estimated at every period of $\Delta T_r = 20 \text{ s}$, and a kinematic control algorithm was used for simulation. It is well known from the literature that a kinematic level control approach can ensure a satisfactory performance when the robot motion demands low speeds and slow accelerations [14]. In this context, we consider that the mobile robot has a low-level control loop with high gain, which imposes any desired velocities $\eta_r = [v_{rx} \ \omega_r]^\top$ to the robot CM such that $u \approx \eta$, wherein $u \in \mathbb{R}^2$ is the control input. From Section II, we obtain the following control system $\dot{q} = S(q)u$. Here, we assume that the control goal is to follow a reference trajectory $q_r(t)$ from the SSMR current configuration q in order to ensure the convergence of the tracking error $e := q - q_r$ to a residual set such that $\lim_{t \rightarrow \infty} \|e(t)\| \leq \epsilon$, where ϵ denotes an arbitrary small positive constant. In this work, we employ a combination of the control strategies introduced by [5], [23], which consider the chained form to obtain the so-called nonholonomic integrator system with drift. The key idea of transforming the kinematic model of a mobile robot in a canonical form is of great interest for solving planning and control problems in an efficient and systematic manner [14], [24]. All numerical simulations have been successfully

performed since there was no collision between the mobile robot and the plant row. In a given tracking task (Fig. 16), the robot firstly maneuvers to enter the corridor, and then drives along it with $v_{rx} = 0.35 \text{ m s}^{-1}$. Notice that, the generated inner path is composed of different straight lines with different yaw angles.



Fig. 16. Simulation: strobe pictures of robot tracking a defined trajectory with different strobe periods T . The utilized parameters for control and reference were [5]: $v_{rx} = 0.35 \text{ m/s}$ (in linear tracking), $\omega_r = 0.5^\circ/\text{s}$ (in rotations), $a = b = 29 \text{ mm}$, $c = 25.5 \text{ mm}$, $r = 20 \text{ mm}$ (wheel radius), $x_{irc} = 0 \text{ mm}$, $\alpha_0 = 1$, $\alpha_1 = 0.5$, $\epsilon_1 = 0.01$, $k_1 = k_2 = 50$, $z_{d1}(0) = z_{d2}(0) = 0.45$.

V. CONCLUDING REMARKS

In this work, we developed a thermal image-based navigation methodology for mobile robots operating in sugarcane crops. The key idea was to combine different research topics such as kinematic modeling approach, robot localization technique, trajectory planning algorithm and motion control strategy in order to create a promising autonomous navigation solution to ensure the successful execution of monitoring tasks in agricultural fields. The IR images provide easy-to-detect features, which are recognized with higher reliability than from the matching RGB ones. More field data should be collected to help us better understand the environment thermal profile under distinct daylight/weather conditions (e.g., night, rain, fog, seasons) and mapping combinations of poor thermal contrast between soil and crop. CT for image segmentation is straightforward to be tuned for most of the found conditions. In spite of big training dataset demand, CNNs have great appeal to deal with the possible situations at the sugarcane field, and strong potential for future investigation to improve inference accuracy and reduce computational cost.

In future works, we will also consider: (i) grayscale IR media without RGB overlay, which requires smaller CNN kernels and may enhance the training process for seeking textures rather than color difference; (ii) conversion to other color spaces such as OHTA [25] which is widely used for fruit detection; (iii) Multi-way classifier for more feature classes, such as ‘pose’, ‘rain/daytime/plant height’, ‘approaching collision’, ‘curved corridor’, etc.; (iv) combination of adaptive control strategies with reinforcement learning approaches to deal with parametric uncertainties. Moreover, we intend to fully integrate the IR camera to the robot control system, and consider the robust dynamic control approach to cope with external disturbances. A strategy for switching crop rows should also be investigated, together with the combination of other inexpensive sensing systems for more accurate results, e.g., a sun position sensor, which is relatively cheap and easy to install, could be employed to provide corrective measures for the robot motion during the navigation phase [26].

REFERENCES

- [1] K. R. Krishna, *Push Button Agriculture: Robotics, Drones, Satellite-Guided Soil and Crop Management*. Apple Academic Press, 2016.
- [2] H. Mousazadeh, "A Technical Review on Navigation Systems of Agricultural Autonomous Off-road Vehicles," *Journal of Terramechanics*, vol. 50, no. 3, pp. 211–232, 2013.
- [3] M. H. Hebert, C. E. Thorpe, and A. Stentz, *Intelligent Unmanned Ground Vehicles: Autonomous Navigation Research at Carnegie Mellon*. Springer Science & Business Media, 2012.
- [4] M. F. S. Xaud, A. C. Leite, E. S. Barbosa, H. D. Faria, G. S. M. Loureiro, and P. J. From, "Robotic Tankette for Intelligent BioEnergy Agriculture: Design, Development and Field Tests," in *XXII Brazilian Conference on Automation*, João Pessoa (PB), Brazil, 2018.
- [5] K. Kozłowski and D. Pazderski, "Modeling and Control of a 4-wheel Skid-Steering Mobile robot," *International Journal of Applied Mathematics and Computer Science*, vol. 14, pp. 477–496, 2004.
- [6] N. S. Krishna, A. George, A. John, and A. Sudheer, "Controller Design for a Skid-Steered Robot and Mapping for Surveillance Applications," in *Proceedings of the Advances in Robotics*, vol. 23, New Delhi, India, 2017, pp. 1–7.
- [7] W. L. Fehلمان and M. K. Hinders, *Mobile Robot Navigation with Intelligent Infrared Image Interpretation*. Springer-Verlag London, 2009.
- [8] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, <http://www.deeplearningbook.org>.
- [9] P. Lottes, J. Behley, N. Chebrolu, A. Milioto, and C. Stachniss, "Joint Stem Detection and Crop-Weed Classification for Plant-specific Treatment in Precision Farming," in *arXiv preprint arXiv:1806.03413*, 2018.
- [10] M. Wedberg, "Detecting Rails in Images from a Train-Mounted Thermal Camera using a Convolutional Neural Network," in *Master of Science Thesis in Electrical Engineering Department of Electrical Engineering, Linköping University*, 2017.
- [11] O. Janssens, R. Van de Walle, M. Loccufier, and S. Van Hoecke, "Deep Learning for Infrared Thermal Image Based Machine Health Monitoring," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 1, pp. 151–159, 2018.
- [12] P. Christiansen, K. A. Steen, R. N. Jørgensen, and H. Karstoft, "Automated Detection and Recognition of Wildlife using Thermal Cameras," *Sensors*, vol. 14, no. 8, pp. 13 778–13 793, 2014.
- [13] L. Deng, M. Yang, Y. Qian, C. Wang, and B. Wang, "CNN based Semantic Segmentation for Urban Traffic Scenes using Fisheye Camera," in *IEEE Intelligent Vehicles Symposium (IV)*, Los Angeles, CA, USA, 2017, pp. 231–236.
- [14] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control*. Springer Science & Business, 2010.
- [15] J. Long, E. Shelhamer, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, 2015, pp. 3431–3440.
- [16] H. Noh, S. Hong, and B. Han, "Learning Deconvolution Network for Semantic Segmentation," in *IEEE International Conference on Computer Vision*, Washington, DC, USA, 2015, pp. 1520–1528.
- [17] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFS," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2018.
- [18] P. Wang, Y. Gan, P. Shui, F. Yu, Y. Zhang, S. Chen, and Z. Sun, "3D Shape Segmentation via Shape Fully Convolutional Networks," *Computers & Graphics*, vol. 70, pp. 128–139, 2018.
- [19] Y.-T. Chen, X. Liu, and M.-H. Yang, "Multi-instance Object Segmentation with Occlusion Handling," in *IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, 2015, pp. 3470–3478.
- [20] B. Chandler and E. Mingolla, "Mitigation of Effects of Occlusion on Object Recognition with Deep Neural Networks through Low-level Image Completion," *Computational Intelligence and Neuroscience*, vol. 2016, no. 13, 2016.
- [21] E. Borenstein and J. Malik, "Shape Guided Object Segmentation," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, New York, NY, USA, 2006, pp. 969–976.
- [22] P. Corke, *Robotics, Vision and Control: Fundamental Algorithms In MATLAB® Second, Completely Revised*. Springer International Publishing, 2017, vol. 118.
- [23] W. E. Dixon, D. M. Dawson, E. Zergeroglu, and A. Behal, *Nonlinear Control of Wheeled Mobile Robots*. Springer-Verlag London, 2001.
- [24] R. W. Brockett, "Asymptotic Stability and Feedback Stabilization," *Differential Geometric Control Theory*, pp. 181–191, 1983.
- [25] P. I. Vasthi and R. Kusumaningrum, "Object Segmentation for Fruit Images using OHTA Colour Space and Cascade Threshold," in *International Conference on Science in Information Technology*, Yogyakarta, Indonesia, 2015, pp. 321–325.
- [26] P. Furgale, J. Enright, and T. Barfoot, "Sun Sensor Navigation for Planetary Rovers: Theory and Field Testing," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 3, pp. 1631–1647, 2011.