

# Hacia una Detección de Surcos Utilizando Visión y Odometría de Ruedas

1<sup>st</sup> Pablo Alonso

*Departamento de Ciencias de la Computación*

*FCEIA-UNR*

Rosario, Argentina

pabloalonsoxx@gmail.com

2<sup>nd</sup> Ernesto Kofman

*CIFASIS*

*CONICET-UNR*

Rosario, Argentina

kofman@cifasis-conicet.gov.ar

3<sup>rd</sup> Taihú Pire

*CIFASIS*

*CONICET-UNR*

Rosario, Argentina

pire@cifasis-conicet.gov.ar

**Resumen**—La robótica agrícola viene siendo impulsada en los últimos años como consecuencia de los avances en las distintas áreas que la componen y de la necesidad de optimizar el trabajo rural. Con el fin de automatizar las actividades agrícolas resulta indispensable que los robots autónomos sean capaces de circular por el campo sin pisar o dañar el cultivo. En general, los cultivos se caracterizan por tener una estructura lineal y se disponen en paralelo, formando hileras, por lo que los métodos de navegación deben poder encontrar estos patrones y guiar al robot por los surcos (espacios entre hileras de cultivo), permitiendo al robot circular a través del campo sin dañar el cultivo.

Este trabajo presenta un método de detección de surcos basado en visión y odometría de ruedas. El mismo consiste en identificar las hileras de cultivo de manera de utilizarlas para la navegación a través de los surcos del campo. El método desarrollado es validado tanto en secuencias de simulación como secuencias reales en un campo de soja. Los resultados muestran que el método es suficientemente robusto como para evitar detectar surcos distintos a los que se venían siguiendo aun cuando el robot se desvíe de los mismos.

**Keywords**—Detección de Surcos, Navegación Autónoma, Odometría de Ruedas, Agricultura de Precisión, Robótica agrícola

## I. INTRODUCCIÓN

A medida que pasa el tiempo, el crecimiento poblacional en el mundo requiere una mayor producción de alimentos. El uso de robots autónomos para realizar tareas agrícolas como siembra, cosecha, remoción de maleza, control de pestes entre otros, resulta una propuesta atractiva debido a que permiten aumentar la productividad haciendo un uso eficiente de los recursos al mismo tiempo que reducen el impacto ambiental en el campo [1], [2].

Asimismo, dado que el campo agrícola presenta un terreno irregular, es visualmente repetitivo y los robots deben recorrer grandes extensiones, el desarrollo de un método de navegación que permita a un robot recorrer el campo de manera segura presenta grandes desafíos [3]–[5]. Una de las técnicas más utilizadas para navegar en el campo agrícola es la de detección de surcos (muchas veces realizada detectando la hileras de cultivos que forman el surco) [6]. Este tipo de sistemas tiene la ventaja que el robot puede navegar grandes distancia independientemente del largo del campo. Los métodos de detección de cultivo usualmente utilizan visión para detectar los patrones que se observan en las hileras de cultivo aunque

pueden complementarse con el uso de otros sensores como GNSS, IMU u odometría. En este trabajo se presenta un método de detección de surcos basado en información visual y de odometría de ruedas. El mismo permite que un robot agrícola pueda navegar por los surcos del cultivo de manera robusta. El método robustece el sistema de detección de surcos propuesto en [6] incorporando información de odometría de ruedas. La odometría de ruedas es utilizada para predecir la ubicación de las hileras de cultivo que rodean al surco de referencia en la imagen actual capturada por la cámara del robot. Esta predicción permite corregir los casos en que el robot se desvíe del surco por el que esta navegando y, mas aun, permite que la navegación vuelva a los surcos aun si la perspectiva de la cámara se aleja demasiado de los surcos. El método es validado utilizando datos de simulación y datos reales capturados por el prototipo de robot desmalezador desarrollado en el instituto CIFASIS (CONICET-UNR).

## II. TRABAJO RELACIONADO

En la literatura se encuentran diversos métodos de detección de surcos basado en visión. Estos métodos en general hacen suposiciones sobre el nivel de verde en los píxeles de la imagen y cómo estos están agrupados, el cultivo está dispuesto en líneas rectas paralelas, y la altura del cultivo, entre otras características. Una de la técnicas utilizada para detectar las hileras de los cultivos es mediante la transformada de Hough (HT, *Hough Transform*). HT es un algoritmo clásico de detección de líneas cuya idea básica es mapear un conjunto de puntos en una imagen a un conjunto de líneas en un espacio de parámetros. La principal desventaja es la cantidad de computo requerido, además de tener mucha redundancia. Rovira et al. [7] definen una ROI (*Region of Interest*) y un *midpoint encoder* para reducir el tamaño de la imagen y aplicar la transformada de HT. Luego, en [8] se adapta HT para que sea capaz de trabajar con plantas pequeñas, mostrando ser efectivo en tiempo real. Ji et al. [9] utilizan una variante denominada transformada de Hough aleatoria (RHT, *Random Hough Transform*) que reduce la cantidad de computo requerido seleccionando puntos en la imagen de manera aleatoria.

Por otro lado, Jiang et al. [10] estiman los centros de las hileras de cultivo mediante una ROI deslizante. Para esto se estiman el principio y fin de cada hilera de cultivo mediante

una función que mide la cantidad de verde en la ROI. La hipótesis es que el nivel de verde aumenta en el comienzo de la hilera, alcanzando su punto máximo en el centro de la misma y decreciente en el final de la misma, por lo que el centro de la hilera debe ser un punto medio entre el comienzo y el fin. Este método además permite estimar el espacio entre hileras y el ancho de las mismas. Por su parte, Zhou et al. [11] plantea que solo es necesario encontrar 2 hileras y que la línea de navegación es entre medio de estas 2 hileras. Para esto define un *set point*  $Q$ , divide la imagen en  $n$  sub-ROIs y por cada sub-ROIs obtiene un conjunto de puntos candidatos los cuales agrega a  $Q$ . Por medio de una regresión lineal, encuentra la línea de navegación como la curva que mejor ajusta los puntos obtenidos en  $Q$ . Otros métodos de detección de surcos, por ejemplo, utilizan visión estéreo para obtener información espacial de los cultivos. En [12] se utiliza un mapa de elevación para detectar la altura de los cultivos y ajustar una función cosenoidal que identifique los surcos entre las hileras de cultivos.

En [13] se asume que las hileras tienen una geometría perfecta y se busca el modelo que mejor las representa. Para esto se proyecta el modelo sobre la imagen. Esta proyección determina una máscara sobre la imagen de las hileras y se encuentra el valor de *yaw* que mejor hace ajustar la máscara sobre las hileras. Luego, se calcula el desplazamiento de las hileras encontrando, mediante regresión lineal, la recta que mejor ajusta a los píxeles en la máscara que corresponden a píxeles sobre las hileras.

### III. DESCRIPCIÓN DEL MÉTODO

El método propuesto es una continuación del método desarrollado en [6], donde se describe un método para estimar  $t$  (en metros) y  $\theta$  (en grados), la distancia y orientación del surco de referencia con respecto al eje delantero del robot. Este método solo requiere una cámara monocular montada en la parte frontal del robot orientada hacia el cultivo. El método [6] se puede resumir en los siguientes pasos:

1. Seleccionar región de interés.
2. Segmentar la imagen.
3. Estimación de  $t - 1, \theta - 1$ .
4. Obtener  $\hat{t}, \hat{\theta}$  del modelo que mejor ajusta a partir de  $t - 1, \theta - 1$ .

Observar que al momento de procesar la imagen  $n$ ,  $t - 1, \theta - 1$  hacen referencia a los valores obtenidos en la imagen  $n - 1$ , siendo  $t = 0, \theta = 90$  los valores iniciales de búsqueda. El método resulta efectivo para detectar las hileras en una imagen y guiar al robot hacia el surco de referencia, pero no demuestra ser lo suficientemente robusto como para evitar saltos de surcos. Se considera salto de surcos cuando el robot detecta hileras distintas a las que venía siguiendo. Esto puede suceder si por algún motivo, por ejemplo, por errores en el control, el robot se desplaza a un surco adyacente al surco que venía transitando (surco de referencia). Esto hace que las hileras de cultivo que se encuentran frente a la cámara no sean las que se venían siguiendo, haciendo que el método seleccione nuevas hileras de cultivo para seguir.

Se plantea que el problema de [6] reside en el paso de estimación de  $t - 1, \theta - 1$ , los cuales no tienen en cuenta el desplazamiento del robot entre las imágenes  $n$  y  $n - 1$ . El problema se percibe especialmente cuando el robot se mueve hacia los costados y el método llega a los límites de su espacio de búsqueda. En esas condiciones  $ty\theta$  toman los valores extremos del espacio de búsqueda arrastrando el modelo.

En la Figura 1 se muestra cómo el método propuesto en [6] salta de surco.

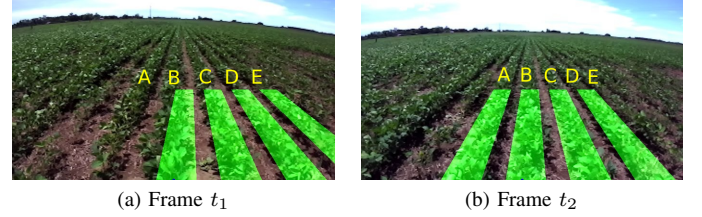


Figura 1. Limitaciones del método de Terissi et al. [6] sobre el cuál se desarrolló el método aquí presentado. Los “saltos” en las detecciones ocurren cuando el robot se mueve hacia la izquierda.

El objetivo de este trabajo es robustecer el método de seguimiento de surcos previamente desarrollado en [6] para que el robot pueda navegar correctamente por el campo agrícola (sin saltar de un surco al otro). Dada una imagen, en [6] se proyecta un modelo de hileras y se estima la rotación ( $\theta$ ) y la traslación ( $\tau$ ) relativos del robot con respecto al surco de referencia. Para esto se determina un conjunto de modelos 3D parametrizados por distintos valores de  $\theta$  y  $\tau$  ( $P_{\theta_1, \tau_1}, P_{\theta_1, \tau_2}, P_{\theta_1, \tau_3}, \dots, P_{\theta_k, \tau_n}$ ). Entonces se elijen  $\theta \in \{\theta_1, \theta_2, \theta_3, \dots, \theta_k\}$  y  $\tau \in \{\tau_1, \tau_2, \tau_3, \dots, \tau_n\}$  tal que la correlación entre  $P_{\theta, \tau}$  y la imagen es máxima entre todo el conjunto de modelos.

Resulta sumamente importante que el método resulte robusto ante saltos, dado que existen diversos motivos por los cuales el robot puede desviar su trayectoria a través de los surcos: irregularidades del suelo, imperfecciones en las ruedas, errores en el control, o una evasión de obstáculos requerida por el sistema de navegación, entre otros. En estos casos el detector debe ser capaz de volver a guiar al robot hacia las mismas hileras que venía siguiendo. Para esto se construye un mapa ( $M_H$ ) de hileras en el marco de *base link* del robot, que contenga la última localización conocida de las hileras y permita predecir  $\theta$  y  $\tau$ . Estos valores predichos de  $\theta$  y  $\tau$  ( $\theta_{pred}$  y  $\tau_{pred}$ ) serán usados como valor inicial en la fase de ajuste de detección. Luego, si la perspectiva de la cámara lo permite, se ajusta la detección desde la pose actual, utilizando un espacio de búsqueda entorno a  $\theta_{pred}$  y  $\tau_{pred}$  hasta encontrar nuevos valores de  $\theta'$  y  $\tau'$  cuyo modelo  $P_{\theta', \tau'}$  es el que mejor ajusta en la nueva imagen como se menciona anteriormente. Si la perspectiva de la cámara no permite que se pueda llevar a cabo el ajuste por un periodo de tiempo, se mantiene temporalmente la posición relativa entre el robot y las hileras seleccionadas, permitiendo al sistema de control volver a los surcos de referencia. Para actualizar  $M_H$  se necesita predecir la pose

del robot en el momento  $t$  ( $R_t$ ) Para predecir la pose  $R_t$  se utiliza la odometría de ruedas.

Entonces, dada la pose  $R_t$ , el mapa  $M_{H_t}$  y  $R_{t+1}$  es posible calcular  $M_{H_{t+1}}$  mediante la transformación relativa ( $R_{t+1}T_{R_t}$ ), la cual transforma puntos desde el marco de referencia en  $R_t$  al marco de referencia en  $R_{t+1}$ . Pese al error propio de odometría, el método propuesto muestra ser efectivo. En la

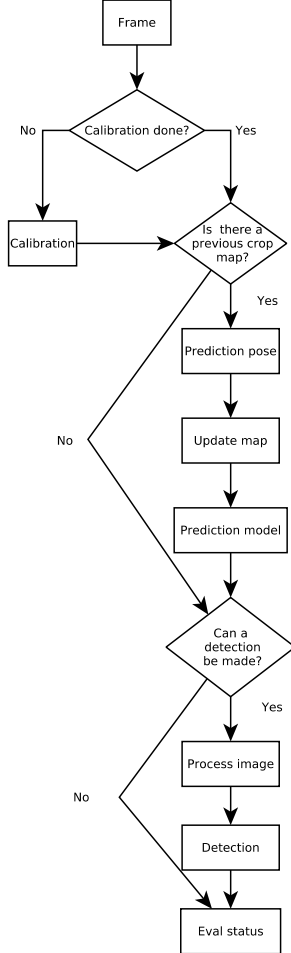


Figura 2. Diagrama del método propuesto.

Figura 2 se muestra el diagrama del método propuesto.

### III-A. Calibración del modelo

El modelo de surcos tiene 6 parámetros: Cantidad de surcos ( $N_S$ ), Separación entre surcos ( $\Delta_S$ ), Ancho de los surcos ( $W_S$ ), Alto de los surcos ( $H_S$ ), Traslación con respecto al robot ( $\tau$ ) y Rotación con respecto al robot ( $\theta$ ).  $N_S$  y  $\Delta_S$  se configuran al inicio del método teniendo en cuenta el cultivo,  $W_S$  y  $H_S$  se estiman exhaustivamente durante el proceso de calibración.  $\theta$  y  $\tau$  son estimados por el método. La salida del método para cada imagen es la traslación y rotación del centro de los surcos detectados con respecto al eje delantero del robot y los comunica al módulo de control del robot para que este siga el surco. En la Figura 3 se muestran los parámetros.

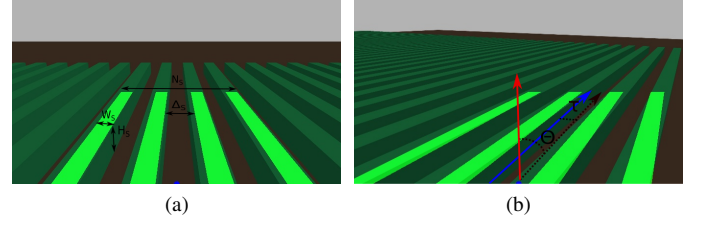


Figura 3. Parámetros del modelo de surcos. (a) se muestran los parámetros de separación entre surcos, altura y ancho de los surcos. (b) la flecha azul indica el centro y la orientación del modelo de surcos con respecto al robot y la flecha roja es el eje  $x$  del robot.

### III-B. Predicción de Pose

Dadas  $v_t$  y  $\omega_t$ , la velocidad lineal y angular del robot en el momento  $t$  respectivamente, sea  $\Delta_t = t_{i+1} - t_i$  el tiempo transcurrido entre dos mediciones reportadas por los *encoders* de los motores del robot. Luego, se predice la pose del robot siguiendo las ecuaciones descriptas en [14]:

$$\begin{aligned}\hat{\psi}_t &= \hat{\psi}_{t-1} + \Delta_t \\ \hat{x}_t &= \hat{x}_{t-1} + v_{t_x} \cos(\hat{\psi}_t) \Delta_t \\ \hat{y}_t &= \hat{y}_{t-1} + v_{t_y} \sin(\hat{\psi}_t) \Delta_t.\end{aligned}$$

### III-C. Actualizar mapa

$M_H^{R_n}$  es un mapa con la ubicación de las hileras dentro del marco  $R_n$ .

La actualización del mapa consiste en obtener un nuevo mapa  $M_H^{R_{n+1}}$  donde los puntos de los surcos se posicionan en el marco de referencia  $R_{n+1}$  desde donde se observa la imagen  $n + 1$ . Para esto se define la transformación relativa  $R_{n+1}T_{R_n}$ , la cual mapea puntos en  $R_n$  a puntos en  $R_{n+1}$ . Al aplicar esta transformación a  $M_H^{R_n}$ , se obtiene  $M_H^{R_{n+1}}$  con los surcos localizados en el marco  $R_{n+1}$ .

### III-D. Predicción de modelo

En esta etapa se predicen los valores de  $\theta_{pred}$ ,  $\tau_{pred}$  utilizados posteriormente para hacer el ajuste de modelo. Figura 4 describe el proceso de predicción.

En la Figura 4a el método procesa la imagen  $k$  y localiza las hileras obteniendo un mapa  $M_H^k$  con la localización de las mismas.

En la Figura 4b se observa el estado del robot luego de desplazarse por lo que posee una pose  $R_{k+1}$ . Cuando se obtiene el frame  $k + 1$ , primero se obtiene  $M_H^{k'}$  actualizando  $M_H^k$  a partir de  $R_{k+1}$ .

En la Figura 4c se predice  $\theta_{pred}$  y  $\tau_{pred}$  a partir de la recta que mejor ajusta  $M_S^{k'}$  para intentar localizar los surcos y obtener  $M_S^{k+1}$ .

### III-E. Procesamiento de la imagen

Como paso previo a la detección, se transforma la imagen de manera que sea mas fácil determinar las zonas de vegetación y no vegetación. Primero se recorta la zona de interés para el análisis (*crop roi*) que corresponde a la parte inferior de

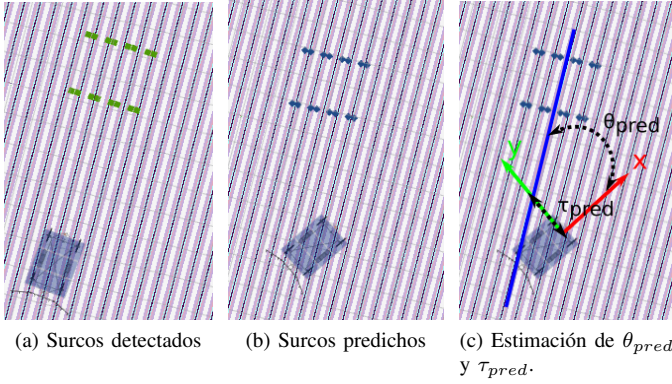


Figura 4. En (a) el método localiza las hileras obteniendo un mapa  $M_H^k$  (puntos verdes). En (b), el robot se desplaza y remapea  $M_H^k$  a partir de su nueva pose obteniendo  $M_H^{k'}$  (puntos azules). En (c)  $\theta_{pred}$  y  $\tau_{pred}$  se calculan a partir de la recta que mejor ajusta  $M_H^{k'}$  y pasa por el eje  $y$  del robot.

la imagen entorno al centro de la cámara. Luego se filtra el verde de la misma utilizando el índice de exceso de verde, el cual se calcula como

$$ExG = 2g - r - b,$$

donde  $r$ ,  $g$  y  $b$  son las componentes cromáticas de cada píxel. Al aplicar esta transformación a una imagen a color se obtiene una nueva imagen en escala de grises donde las zonas con vegetación resultan con mayor valor de intensidad que las zonas de suelo. Luego, mediante la aplicación de un umbral se obtiene una imagen binaria que indica que píxeles pertenecen a cada caso. La aplicación del umbral se realiza mediante el método de Otsu [15]. En la Figura 5a se observa la imagen que recibe el método como entrada y en la Figura 5b el estado de la misma antes del paso de detección.

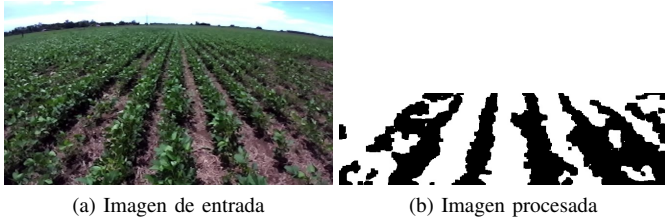


Figura 5. (a) imagen de entrada del método obtenida por la cámara. (b) Imagen binaria obtenida después del *crop roi* y la umbralización.

### III-F. Detección

El resultado de haber procesado la imagen como se describió en la sección anterior se denomina  $I_G$ .  $I_G$  es una imagen binarizada, matemáticamente una matriz donde la entrada  $i, j$  corresponde a 1 para verde y 0 para el resto. El proceso de detección consiste en hacer una búsqueda de  $\theta$  y  $\tau$  de manera que el modelo de surcos obtenido ( $P_{\theta, \tau}$ ) sea el que mejor ajuste a  $I_G$ .

**III-F1. Crear espacio de búsqueda:** Para encontrar  $\theta$  y  $\tau$  adecuados a la imagen dada, se hace una búsqueda exhaustiva en un espacio de búsqueda dado ( $S$ ), el cual se define a

partir del estado actual del detector. Como se detalla mas adelante, el detector tiene 2 estados: *detección* o *no detección*. Si el sistema está en el estado *detección*, existe  $M_{H_k}^k$  con la localización de las hileras y  $\theta_{pred}$  y  $\tau_{pred}$  se calculan como ya se mencionó. Luego, se crea un espacio de búsqueda entorno a estos valores, es decir se define  $\delta_\theta$  y  $\delta_\tau$ , luego:

$$S = \{\theta, \tau |$$

$$\theta_{pred} - \delta_\theta \geq \theta \geq \theta_{pred} + \delta_\theta, \tau_{pred} - \delta_\tau \geq \tau \geq \tau_{pred} + \delta_\tau\}$$

En el estado de *no detección*, se hará una búsqueda de  $\theta \in [0, 180]$  en y  $\tau \in [-1, 1]$ , donde  $\tau = -1$  corresponde al extremo derecho de la imagen y  $\tau = 1$  al extremo izquierdo. Es decir, en el estado *no detección* se hace un barrido completo de la imagen.

**III-F2. Calculo de correlación:** La correlación estima que tan bien ajusta  $P_{\theta, \tau}$  a  $I_G$  de acuerdo a la cantidad de verde en los surcos y la zona de tierra o entre surcos. Para esto se proyectan máscaras sobre  $I_G$  que determinan la cantidad de verde en una zona de la imagen:

- $I_{P_{\theta, \tau}}^H$ , máscara de verde sobre la zona de surcos.
- $I_{P_{\theta, \tau}}^T$ , máscara de verde sobre la zona de tierra.

Observar que las máscaras quedan parametrizadas únicamente por  $\theta$  y  $\tau$ . Dado que tanto las máscaras como la imagen son matrices binarias, donde la entrada  $i, j$  tiene 1 para verde y 0 para el resto, matemáticamente se puede definir una función de costo que calcule el nivel de vegetación en la zona de la imagen. Dado  $\theta$  y  $\tau$ , la correlación de la zona de surcos se define como:

$$D_S(I_G, I_{P_{\theta, \tau}}^S) = \frac{\sum_{i,j} I_G(i,j) I_{P_{\theta, \tau}}^S(i,j)}{\sum_{i,j} I_{P_{\theta, \tau}}^S(i,j)}$$

Análogamente, la correlación en la zona de tierra se define:

$$D_T(I_G, I_{P_{\theta, \tau}}^T) = \frac{\sum_{i,j} I_G(i,j) I_{P_{\theta, \tau}}^T(i,j)}{\sum_{i,j} I_{P_{\theta, \tau}}^T(i,j)},$$

Ambas funciones de costo toman valores en el rango  $[0, 1]$ . Observar que cuando el modelo ajusta en una zona de surcos con mucha vegetación,  $D_H \approx 1$ , y con poca vegetación en la zona de tierra  $D_T \approx 0$ . Luego se define:

$$C(I_G, \theta, \tau) = D_H(I_G, I_{P_{\theta, \tau}}^H) - D_T(I_G, I_{P_{\theta, \tau}}^T),$$

cuyo valor en  $[-1, 1]$  indica que tan bien ajusta  $P_{\theta, \tau}$  en la imagen. Luego, la rotación y la traslación de los surcos con respecto a la cámara están dadas por  $\hat{\theta}$  y  $\hat{\tau}$  tales que:

$$\hat{\theta}, \hat{\tau} = \max_{(\theta, \tau) \in S} \{C(I_G, \theta, \tau)\}.$$

Entonces, lo que se intenta localizar en la imagen es la región con mayor verde en la zona de surcos y menor cantidad de verde en la zona de tierra. Además se define  $c_f$  como el umbral de correlación, luego si  $C(I_G, \hat{\theta}, \hat{\tau}) > c_f$ , se considera que la detección fue exitosa.



### III-G. Evaluar estado

Como ya se mencionó, el sistema tiene 2 estados posibles luego de haber procesado  $n$  imágenes:

- *detección*: El robot tiene un mapa  $M_{S_n}$  con la ubicación de los surcos. La búsqueda en este estado es acotada en un entorno cercano a los surcos detectados. En caso de que la perspectiva de la cámara este demasiado alejada de los surcos de referencia y no se puede llevar a cabo la detección, se permanecerá en este estado por una cantidad limitada de frames guiando la navegación solo con la predicción hecha.
- *no detección*: El robot no tiene ningún mapa y la búsqueda sera exhaustiva sobre la imagen.

Para evitar estar pasando continuamente de un estado a otro se requiere un periodo de validación antes de aceptar el nuevo estado. Este periodo se mide en el procesamiento de  $m$  frames y el resultado obtenido por el método. Entonces, partiendo de un estado *no detección*, se requiere  $m$  detecciones consecutivas antes de que el sistema acepte el estado. Análogamente, si no hay detecciones en  $m$  imágenes de manera consecutiva, se vuelve al estado *no detección*.

## IV. EXPERIMENTOS

Para la validación del método propuesto se realizaron experimentos tanto en un entorno de simulación como en un entorno real. Para las pruebas reales, se utiliza una secuencia capturada por el prototipo del robot desmalezador del CIFASIS (CONICET-UNR) [3] en un campo de soja de la localidad de Zaballa, Santa Fe. Las imágenes fueron capturadas por la cámara ZED del robot con resolución de  $672 \times 376$  píxeles y 15 fps. Para las pruebas en simulación se utilizó una secuencia generada con el framework de simulación presentado en [16]. El mismo está realizado con el simulador Gazebo, y consta de un modelo del prototipo del robot desmalezador en un entorno simplificado del campo agrícola (las hileras de cultivo están representadas por paralelepípedo rectangulares de color verde). Para la realización de los experimentos se utilizó una notebook Lenovo P50 con CPU Intel i7-6700HQ de 2,60 GHz, GPU NVIDIA GM107GLM y 30 GB de memoria RAM.

En lo que sigue se analizan los resultados obtenidos por el método propuesto y el método presentado en [6]. En la Figura 6 se observa la ejecución del método de [6] (sin odometría) sobre la secuencia sintética. El método selecciona los surcos B, C, D y E. Al girar el robot hacia la izquierda, la cámara va rotando y el método va prediciendo utilizando los valores de  $\theta$  y  $\tau$  de la última detección realizada, esto causa que el modelo (franja verde) rote junto con la cámara hasta que la predicción caiga en la zona de tierra (franja azul entre los surcos). Visualmente se ve como un “arrastre” de las franjas azules hasta caer en la zona de los surcos A, B, C, D donde se vuelven a detectar hileras. Esto sucede principalmente ya que el método tiene como hipótesis que las hileras en la imagen van a estar muy cerca de su ubicación anterior, sin tener una representación o mapa de las hileras ni una predicción de pose que tiene el robot. En la Figura 7 ilustra

un comportamiento similar en una secuencia real capturada por el robot. Se observa primero la predicción y el ajuste (Figura 7a, Figura 7b) sobre un mismo frame. Posteriormente el robot se desvía hacia la izquierda y se observa un salto en la predicción y ajuste en la Figura 7c y Figura 7d. En este caso el centro de la cámara se aleja tanto de los surcos que venía siguiendo que reinicia la búsqueda al centro de la misma.

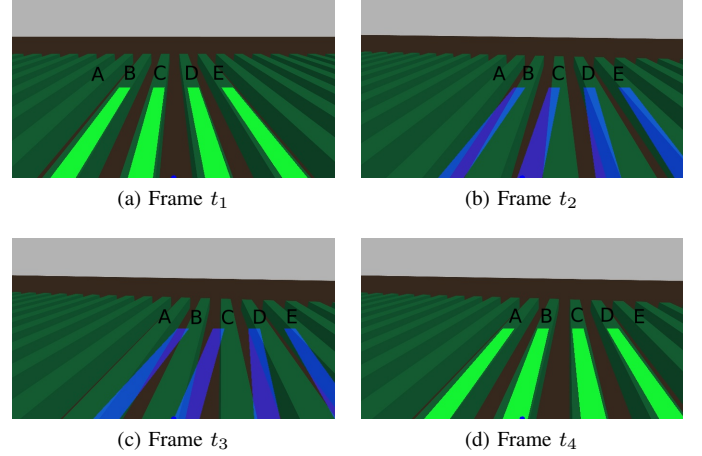


Figura 6. Método de detección de surcos propuesto en [6] sobre la secuencia de simulación.

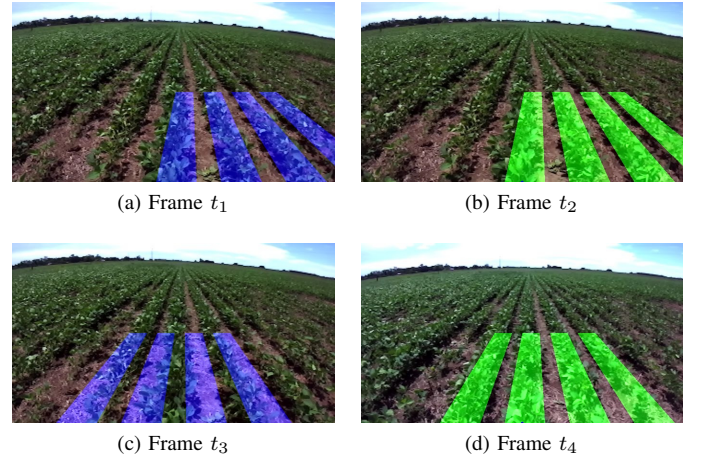


Figura 7. Método de detección de surcos propuesto en [6] sobre la secuencia real.

De manera análoga, en la Figura 8 y Figura 9 se muestra el comportamiento obtenido con el método propuesto en los mismos tramos. En la Figura 8c y Figura 8d se observa que el mapa de hileras de cultivo y la odometría de ruedas permite realizar una predicción que se solapa con las hileras que se venía siguiendo aún cuando la cámara no enfoca directamente las mismas. Esta predicción puede ser utilizada por el sistema de navegación por un breve lapso de tiempo (estado *detección*) hasta que el robot pueda retomar el surco de referencia. Por otro lado, en la Figura 9b se observa que la predicción si bien está cerca de las hileras que se estaban siguiendo no se solapa

completamente. La mala predicción se debe al error propio de la odometría del robot.

Se calcula la mediana de todas las correlaciones obtenidas por las predicciones en los datos reales, con ambos métodos, obteniendo 0.44 con el método propuesto mientras que [6] obtiene 0.34, mostrando que el método propuesto mejora la calidad de la predicción.

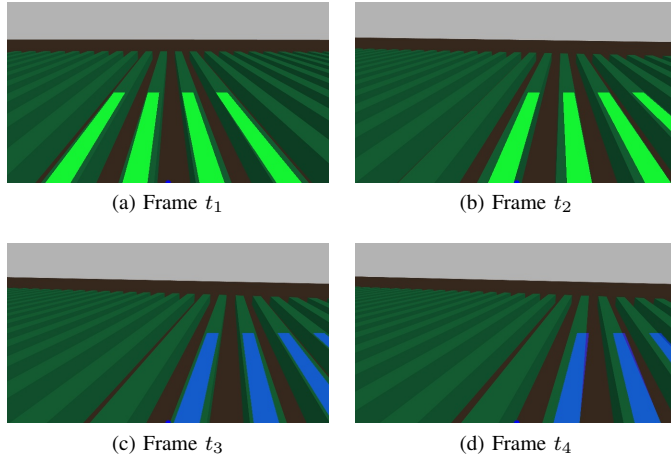


Figura 8. Método de detección de surcos propuesto sobre la secuencia simulada.

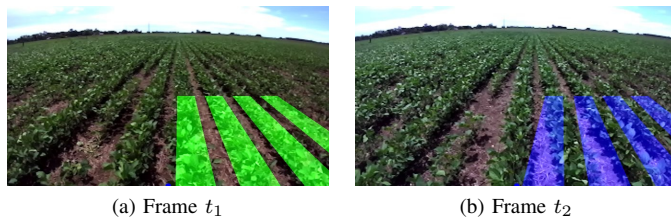


Figura 9. Método de detección de surcos propuesto sobre la secuencia real.

El tiempo de procesamiento para cada *frame* en los datos reales es de 20 ms en caso de que el sistema se encuentre en el estado de *detección* y 28 ms para el estado de *no detección*, duplicando el reportado en [6] pero aun siendo apto para su uso en tiempo real.

## V. CONCLUSIONES Y TRABAJO FUTURO

En este trabajo se presenta un método de detección y seguimiento de surcos basado en información visual y odometría de ruedas. El método utiliza los datos de odometría proveniente de los encoders de las ruedas para predecir dónde serán vistas las hileras de cultivo en la imagen actual capturada por la cámara. A partir de la predicción y el estado del sistema (*detección* o *no detección*) se define un espacio de búsqueda de modelos de hileras, de los cuales se elige el que mejor ajusta en la imagen actual. Los experimentos muestran que el método de detección de surcos (a través de la detección de hileras) es capaz de predecir utilizando odometría de ruedas la ubicación de las hileras de cultivo en el frame actual de la cámara incluso en casos donde el robot se desvía del surco

de referencia. Dicha predicción permite acotar la búsqueda de las hileras en la etapa de detección o redirigir al robot hacia las mismas hileras de referencia ante un desvío.

Como trabajo futuro se planea integrar información de otros sensores como IMU y GNSS para robustecer aún más la etapa de predicción del método.

## AGRADECIMIENTOS

Este trabajo fue apoyado por CONICET (PIBAA N° 0042), AGENCIA I+D+i (PICT 2021-570) y Universidad Nacional de Rosario (PCCT-UNR 80020220600072UR).

## REFERENCIAS

- [1] F. A. Auat Cheein and R. Carelli, "Agricultural Robotics: Unmanned Robotic Service Units in Agricultural Tasks," *IEEE Industrial Electronics Magazine*, vol. 7, no. 3, pp. 48–58, 2013.
- [2] C. W. Bac, E. J. van Henten, J. Hemming, and Y. Edan, "Harvesting Robots for High-value Crops: State-of-the-art Review and Challenges Ahead," *Journal of Field Robotics*, vol. 31, no. 6, pp. 888–911, 2014.
- [3] T. Pire, M. Mujica, J. Civera, and E. Kofman, "The Rosario Dataset: Multisensor Data for Localization and Mapping in Agricultural Environments," *Intl. J. of Robotics Research*, vol. 38, no. 6, pp. 633–641, 2019.
- [4] J. Cremona, R. Comelli, and T. Pire, "Experimental evaluation of Visual-Inertial Odometry systems for arable farming," *Journal of Field Robotics*, vol. 39, no. 7, pp. 1123–1137, 2022.
- [5] J. Cremona, J. Civera, E. Kofman, and T. Pire, "GNSS-stereo-inertial SLAM for arable farming," *Journal of Field Robotics*, vol. n/a, no. n/a, pp. 1–11, 2023.
- [6] L. D. Terissi and G. Sad, "Detección automática de surcos de cultivos a partir de imágenes," in *AADECA 2018 - Semana del Control Automático - Congreso Argentino de Control Automático*, 2018, pp. 1–6.
- [7] F. Rovira-Más, Q. Zhang, J. F. Reid, and J. D. Will, "Hough-transform-based vision algorithm for crop row detection of an automated agricultural vehicle," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 219, no. 8, pp. 999–1010, 2005.
- [8] W. Winterhalter, F. V. Fleckenstein, C. Dornhege, and W. Burgard, "Crop Row Detection on Tiny Plants With the Pattern Hough Transform," *(IEEE) Robotics and Automation Letters*, vol. 3, no. 4, pp. 3394–3401, 2018.
- [9] R. Ji and L. Qi, "Crop-row detection algorithm based on Random Hough Transformation," *Mathematical and Computer Modelling*, vol. 54, no. 3, pp. 1016–1020, 2011, mathematical and Computer Modeling in agriculture (CCTA 2010).
- [10] G. Jiang, Z. Wang, and H. Liu, "Automatic detection of crop rows based on multi-ROIs," *Expert Systems with Applications*, vol. 42, no. 5, pp. 2429–2441, 2015.
- [11] Y. Zhou, Y. Yang, B. Zhang, X. Wen, X. Yue, and L. Chen, "Autonomous detection of crop rows based on adaptive multi-ROI in maize fields," *International Journal of Agricultural and Biological Engineering*, vol. 14, no. 4, pp. 217–225, 2021.
- [12] M. Kise, Q. Zhang, and F. Rovira Más, "A Stereovision-based Crop Row Detection Method for Tractor-automated Guidance," *Biosystems Engineering*, vol. 90, no. 4, pp. 357–367, 2005.
- [13] J. M. Guerrero, J. J. Ruz, and G. Pajares, "Crop rows and weeds detection in maize fields applying a computer vision system based on geometry," *Computers and Electronics in Agriculture*, vol. 142, pp. 461–472, 2017.
- [14] K. S. Chong and L. Kleeman, "Accurate odometry and error modelling for a mobile robot," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, vol. 4, 1997, pp. 2783–2788.
- [15] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Intl. Conference on Systems, Man and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.
- [16] I. Ait, E. Kofman, and T. Pire, "A Travelling Salesman Problem Approach to Efficiently Navigate Crop Row Fields with a Car-Like Robot," *(IEEE) Latin America Transactions*, vol. 21, no. 5, April 2023.