Hand gesture recognition for driver vehicle interaction

Yannick Jacob, Sotiris Manitsaris and Fabien Moutarde
Robotics Laboratory (CAOR), Mines ParisTech, France
{firstname.lastname}@mines-paristech.fr

Gautam Lele and Laetitia Pradere,
PSA Peugeot Citroën, France
{firstname.lastname}@mpsa.com

***Abstract*— In this paper, we present a new driver vehicle interface based on hand gestures that uses a hierarchical model to minimize resources requirements. Depth information is provided by time of flight sensor with automotive certification. In particular, we develop our implementation of a Random Forest based posture classification in two subcases: micro gestures at the wheel and macro gestures in front of the touch screen.**

Keywords: hand gesture recognition, human computer interaction, micro gestures, automotive sensing.

I. CONTEXT AND OBJECTIVES

As modern cars continue to offer more and more functionalities, they require a growing number of commands. In order to be controlled, two main options are usually taken: adding buttons to the already clattered dashboard or adding items in the graphic user interface. Even though these menus do not seem specially complex compared to their smartphone counterpart, they present important flaws when it comes to driver vehicle interaction. One of the main problems lays in their need for visual attention. As the driver tries to monitor the road and the user interface at the same time, his overall efficiency is reduced. In the case of the primary task – driving, the principal consequences are safety issues. The secondary tasks, which involve the control of non-driving related functions and other interactions with the dashboard, should offer better driving conditions.

In order to reduce the visual attention needed for secondary tasks while enabling an easy interaction with the car's functionalities, a change in the paradigm of driver vehicle interaction seems to be needed. While physical buttons, touchscreens and voice recognition are widely used, another potential modality is still often left aside: gestures [1]. It is especially detrimental as in the first place, gestures are needed in order to reach for a button or a touch screen.

In an automotive environment, gestures can be used both as an interaction modality and for monitoring. The first case encompasses the driver performing specific gestures that are mapped to specific command by the dashboard, i.e. replacing traditional user interfaces. Gesture monitoring aims at retrieving information about the driver's current actions. In particular, such information can provide cues about the driver being on the verge of performing a gesture aimed at the dashboard. They can also assess if the driver is maintaining a sufficient level of safety – typically combined with a gaze recognition system.

Wearable systems, used in [2] are not possible to use as we want our system to be usable by anyone entering the car. We therefore will focus on the use of vision systems. We chose to use a sensor using Time of Flight technology that contrary to traditional passive sensors, such as usual cameras, or basic active sensors, is able to work both in obscurity and abundant daylight. Given the price of those sensors, we focus the use of depth map from a single source.

However, the complexity of gestural movements makes them hard to fully recognize. In particular, the very high deformability of the hand and the higher variability of hand poses make that task even harder. Recent papers have demonstrated a growing ability to detect hand pose or infer hand skeleton in an increasing number of cases. Nonetheless, many of them require complex processing, especially when using a model optimisation based approach [3, 4], which is currently unavailable for this type of functionality in a standard automotive dashboard.

II. METHOD

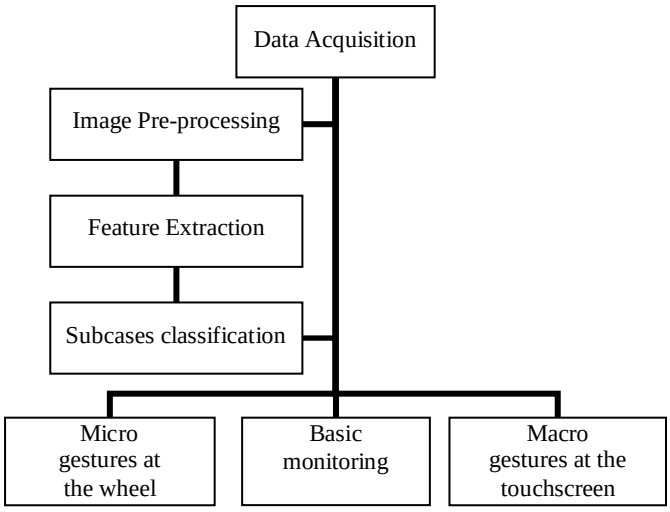
We believe one way to limit the computational power needed for such a task is to use only complete information about the hand when needed. In particular, it might not be useful to know exactly where every hand key-point when the driver is only readjusting his glasses or reaching for his wallet.

We propose the use of a hierarchical model that performs cascaded features extraction. In particular, it provides only the minimal information required at all time. The first layer of the model computes basic features extraction enabling a simple classification. This classification determines if further processing is performed or not, thus saving computational time.

Not only can it reduce overall computational need of the system, this also permits the use of simpler model for each specific subcase, as prior information about the subcase in known. We believe this is specifically justified by the need for a very robust system only in particular subcases, to unsure the best interaction experience for the driver.

On the contrary to complete models that need huge datasets to be estimates the various parameters of their model [5, 6], our method is able to perform well in the desired configuration with a small number of images.

A. *System Architecture*



B. ***First subcase :*** *Palm-grasp micro-gestures at the wheel.*

For safety reasons, hands should hold the wheel while driving. While this guideline is not completely followed, drivers still spend a majority of their time in the car with at least one hand on the wheel. Enabling the driver to interact with the dashboard while keeping his hands on the wheel is therefore seen as an amelioration of the interaction and the safety, provided the gestures are easy to perform and remember.

We are here interested in micro gestures, that is minute finger movements that do not interfere with the driver performing his primary task, driving. We think the area around and above the wheel has a strong potential for interaction with the dashboard. This however leads to some limitation, including the presence of the wheel between the sensor and the hand, which suffer partial occultation. Modelling hand gesture when interacting with an object is particularly challenging.

C. ***Second subcase :*** *Macro gestures in front of the touscreen.*

In order to interact with the touchscreen without touching it, we believe the use of macro gestures in the area in front of it can be useful. They reduce both the need for visual attention and the completion time compared to usual touchscreen interaction.

Large gestures present a potentially high affordance to a wide variety of people. For example, swipe gestures are easily associated with navigation inside a menu. Furthermore, knowing the hand configuration can significantly meliorate a gesture recognition system. For example, it will help discriminate between intentional gestures directed at the dashboard and other gestures. It can also offer variation on gestures depending on the hand posture, just as touch strokes trigger different actions depending on the number of fingers detected.

III. IMPLEMENTATION

A. ***Preprocessing***

After the depth map is acquired via the time of flight sensor, focusing on the region of interest given by the first layer of the system.

The first step is to initialise the position of the wheel relative to the camera. A first estimation is made using a circular hough transform that returns the best fit for circles in the image. If the radius and the distance give coherent results, a further processing is made to assess the exact position and orientation. One of the main needs is to evaluate the angle between the plane of the wheel and the depthmap plane.

In order to account for the rotation of the wheel a transform is performed. It was initially an inverse rotation using the current position of the hand, but both for processing time (need to be recomputed at each frame) and instability; we instead used a custom transformation based on gnomonic projection.

Gnomonic projection is usually used in cartography to project a sphere into a plane while preserving shortest route relation: the shortest geodesic route on a sphere will exactly be the shortest segment on the plane. In particular, each diameter of a sphere will be projected as a line. This property motivated our choice as it permits to transform a rotation in a translation, as moving along the wheel is mapped to moving along a line.

In our case, the representation space can be seen as a Cartesian display of polar coordinates. Considering the 2D plane of the wheel, the Cartesian coordinates of a point are its projection along two axes, the abscissa and the ordinate. We chose to instead take the Polar coordinates - the distance toward the centre of the wheel point and the angle from a fixed reference upward direction. The target representation defines the ordinate as the distance from a fixed point, and the abscissa as the signed distance from the upward direction to the point along circles around the centre of the wheel.

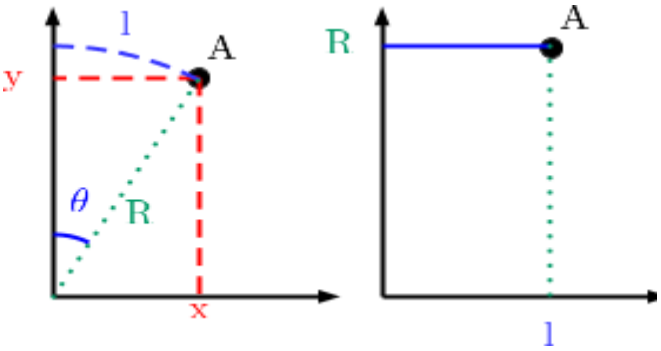


Figure 1. Illustration of the transform of a point A. Left: Cartesiant(x,y) and Polar(R,theta) coordinates. Right: Target domain (l, R)

This transform is thus only dependent on the position of the wheel relative to the camera. As long as the configuration is known, the transform can be computed once for all, thus limiting online processing to constants link between the two spaces.

B. *Pixelwise classification*

The depth images are then classified different labels using a random decision forest algorithm comparable to [5]. The method is inspired by [6, 8]. We are currently comparing the effect of the number of labels used for the classification process including in terms of speed, accuracy, consistency, complexity and robustness.

When using only 6 labels, we only identify the different fingers using a very small forest, encompassing a couple of trees with a depth of a semi dozen. The forest can be efficiently trained using less than a hundred of manually annotated images.

Figure 2. Example of the Random Decision Forest output for pixelwise classification in the case of finger identification.

To ensure the best fingertip location is found, we use a technique comparable to MPI, looking in the X and Z plane for maxima. We believe in our configuration, fingertips will be prominent in one of those directions when they are not resting on the wheel.

Once fingertips are located, they are used to infer relevant information about the hand. In particular, hand pose estimation is done using a heuristic approach based on the distance between fingertips and the wheel. A finite state machine is used to output the current posture.

A working demo has been tested by various car users and a systematic evaluation will be conducted in the upcoming weeks.

Complementary approaches are currently producing encouraging results implementing Dynamic Time Warping and Hidden Markov Models to recognize dynamic gestures, enabling more complex and natural interactions.

IV. CONCLUSIONS AND PERSPECTIVES

We present here our hierarchical system to address the problem of automotive hand gesture recognition for driver vehicle interaction. In particular, we only compute basic features at all time and trigger fine-tuned models for specific subcases only.
The system is currently able to recognize independent gestures made by the finger. Further evaluation is being conducted to assess the performance of the current system, while ameliorations are regularly being added.

Future works include the refinement of the subcase classifier, the evaluation of the influence of the number of classes and the comparison with a unified system, such as [7].
A coarse to fine approach is also being investigated, inspired by [9]

REFERENCES

[1] Pickering, C. A., Burnham, K. J., & Richardson, M. J. (2007, June). A research study of hand gesture recognition technologies and applications for human vehicle interaction. In 3rd Conf. on Automotive Electronics.

[2] Wang, R. Y., and Popovíc, J. Real-time hand-tracking with a color glove. In ACM Trans. Graph., vol. 28 (2009), 63:1–63:8.

[3] Oikonomidis, I., Kyriazis, N., and Argyros, A. Efficient model-based 3D tracking of hand articulations using Kinect. In Proc. BMVC (2011), 1–11.

[4] Xu, C., and Cheng, L. Efficient hand pose estimation from a single depth image. In Proc. ICCV (2013), 3456–3462.

[5] Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., and Blake, A. Real-time human pose recognition in parts from a single depth image. In Proc. CVPR (2011).

[6] Keskin, C., Kirac¸, F., Kara, Y. E., and Akarun, L. Hand pose estimation and hand shape classification using multi-layered randomized decision forests. In Proc. ECCV (2012), 852-863.

[7] Tang, D., Yu, T.-H., and Kim, T.-K. Real-time articulated hand pose estimation using semi-supervised transductive regression forests. In Proc. ICCV (2013).

[8] Dapogny, A., De Charette, R., Manitsaris, S., Moutarde, F., & Glushkova, A. Towards a hand skeletal model for depth images applied to capture music-like finger gestures. In Proc. CMMR (2013).

[9] Tang, D., Chang, H. J., Tejani, A., and Kim, T.-K. Latent regression forest: Structured estimation of 3D articulated hand posture. In Proc. CVPR (2014).