Supplementary material for Instance Invariant Visual Servoing Framework for Part-Aware Autonomous Vehicle Inspection using MAVs

Harit Pandya
IIIT-Hyderabad, Hyderabad, Telangana-500032, India
email:harit.pandya@research.iiit.ac.in

Ayush Gaud
IIIT-Hyderabad, Hyderabad, Telangana-500032, India
email:ayush.gaud@gmail.com

Gourav Kumar
IIIT-Hyderabad, Hyderabad, Telangana-500032, India
email:gourav.kumar@research.iiit.ac.in

K. Madhava Krishna
IIIT-Hyderabad, Hyderabad, Telangana-500032, India
email:mkrishna@iiit.ac.in

1 Gazebo simulation experiments

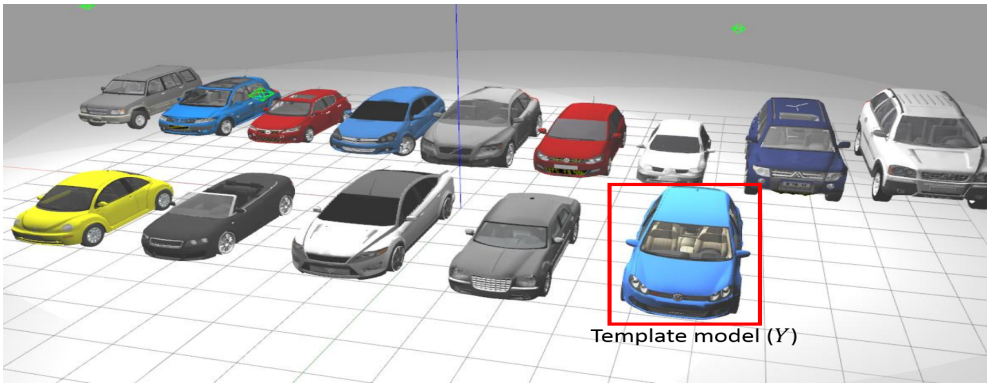

Figure 1: **CAD models used for Gazebo simulation experiments.** The last car in the second row is used as template for all the experiments in this paper while other cars are used as novel instances for validating the approach in Gazebo simulation.

After evaluating the components of our vision pipeline such as keypoint predictions, part segmentation and related refinements individually, we now evaluate our pose induction and navigation module for the entire visual inspection pipeline in Gazebo simulation framework. The setup comprises of a MAV platform, for which we use RotorS Gazebo library, and twelve 3D CAD models of cars as shown in figure 1. We use one car among them as the template model Y and rest are used as previously unseen novel instances X for the inspection. The simulation environment mimics ideal condition with accurate odometry readings from the MAV and no disturbances from external factors like wind.

1.1 Results for instance-invariant PBVS

In this experiment we evaluate the instance-invariant PBVS comprising of pose induction and side-navigation module in Gazebo simulation. The objective is that starting from a provided random pose $\mathcal{F}_0$ and a desired image of a side I_s^*, MAV needs to estimate $\mathcal{F}_s^*$ using the pose induction and reach there using the side-navigation module. We select 12

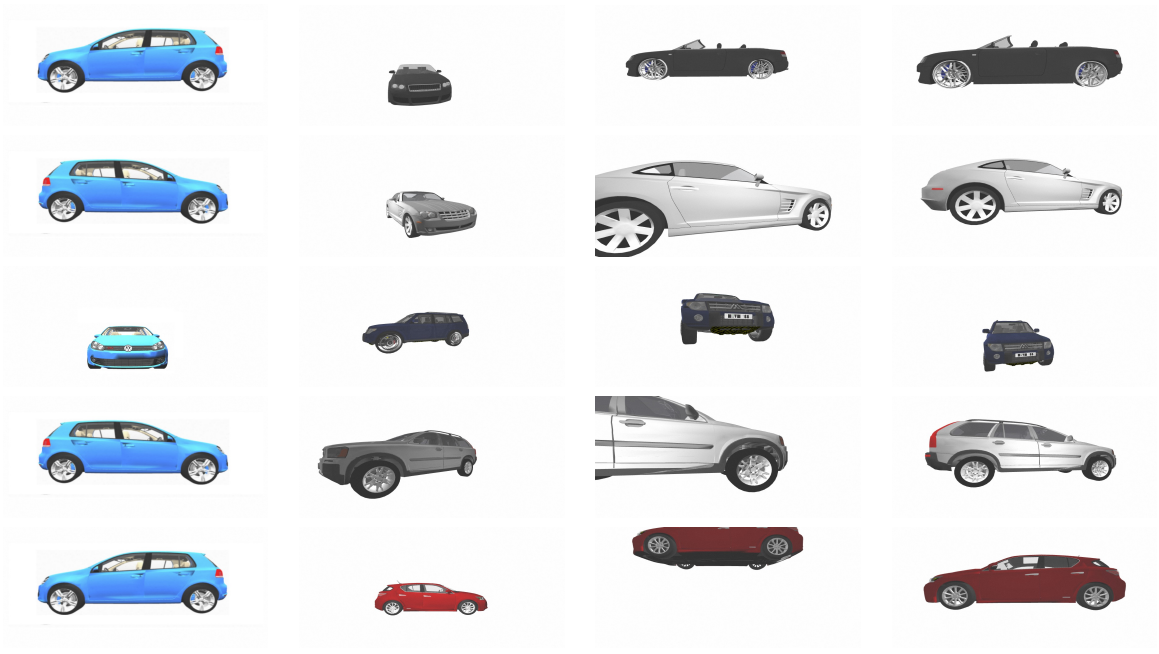

Figure 2: **Qualitative results of instance-invariant PBVS pipeline comprising of pose induction and side-navigation module.** (a) Desired camera pose of the template instance. (b) Initial pose of the current instance captured by the robot from a random initial position. (c) Resultant positioning of camera achieved by instance invariant PBVS approach. (d) Improvement in the resultant positioning of the MAV on account of IBVS refinement. Note the similarity in the resultant pose achieved by the proposed approach compared to the desired pose provided, even for large camera transformations and non-overlapping scenes.

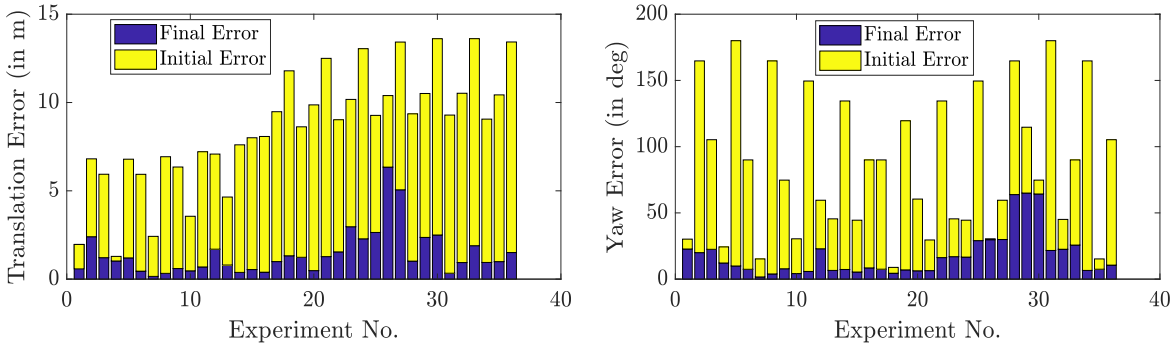

Figure 3: **Quantitative results of instance-invariant PBVS pipeline.** (a) The initial translation error between the initial pose of MAV and desired pose for 12 different instances and 3 pose per instance is compared with the final error resulting from our instance invariant PBVS approach followed by IBVS refinement. (b) Initial and resultant error in yaw. Note our approach is able to perform servoing across 12 different instances using only the template model. Furthermore, our approach can tackle large camera transformation with initial error of over 10 m and even non-overlapping scenes where yaw error is greater than 90 degrees.

models of the car from our simulation dataset and use them as the current instances X. We initially render the template model Y to get the desired views I_s^* corresponding to the sides. Furthermore, the MAV is initialized at a random pose $\mathcal{F}_0$, such that the car X is in the field of view and instance-invariant PBVS is used to servo the MAV to the desired pose for one of the sides $\mathcal{F}_s^*$. The qualitative results of the simulation are shown in figure 2. The first column shows

the desired view I_s^* of the template instance Y. While the second column shows the image captured by MAV from the initial pose $\mathcal{F}_0$ of the current instance X. The third column represents the final view captured by the MAV using only instance invariant PBVS. Finally, the last column showcases the improvement in the performance of final pose attained by the MAV owing to the IBVS refinement. It can be observed that the resultant image after IBVS refinement is close to the provided desired view. Note that although the odometry of the MAV is accurate, there are still errors in the final pose attained by the MAV, this is due to the large variation in shape between X and Y. It can be seen that our approach can easily tackle large variations in shape, texture and camera pose and it even works for non-overlapping scenes, which are challenging scenarios for existing visual servoing approaches.

For quantitatively validating the instance-invariant PBVS, we again consider the 12 models from our synthetic dataset. The MAV is initialized from a random starting point and the objective is to attain the desired pose. Here we place all the models at same location and orientation in the Gazebo world. We further consider 3 desired pose per model which remain constant for all the models so that, we have 12 x 3 experiments in total. For every experiment, we record the initial error which is the between initial pose and the desired pose in translation and yaw. We then use our instance-invariant PBVS framework to navigate the MAV to the desired pose and record the final error which is error between the final pose achieved by our approach and desired pose. It can be seen from figure 3 that our approach is able to converge near the desired pose despite of large initial pose error. Note that since we are servoing across instances there will be a residual error in the final pose.

2 Evaluating the autonomous inspection pipeline

2.1 Evaluating visual inspection

After exhaustive testing of individual components, we next validate the performance of the complete pipeline in Gazebo simulation framework. Here we use 13 CAD models of cars in addition to the template model. The MAV starts with random initial pose and servo to the 8 essential parts (4 wheels, 2 headlights, 2 mirrors) for every car. We do not test on taillights because taillights are not part of PASCAL part dataset. With sufficient training sample our approach could easily be extended to other parts or object categories. Here we assume that the workspace is obstacle free and car is visible in the initial random pose. The qualitative results for excluding and including Bayesian refinement are shown in figures 4 and 5 respectively. These results display the final images captured after servoing to a individual parts overlaid with their segmentation masks computed by our network. It could be seen that with our approach the MAV is able to reach the correct parts despite different shapes of essential parts and their configurations. The quantitative results are designed to measure the servoing performance on resultant values of area. Table 1 shows the area in percentage of image size (856x480 px) achieved after servoing to a specific part. The chosen desired area for wheels is 10 % and for mirrors and lights is 2 % of the total area of the image. The table also highlights the improvement in performance due to the multi-view Bayesian fusion. It can be seen that with the fusion the mean of resultant areas are nearer to their desired values. Furthermore, the number of instances where the visual servoing diverges (marked by '-' in the table) are also less after Bayesian fusion.

2.2 Evaluating visual servoing

Once we have evaluated our approach on visual inspection parameters, we now show the performance of our approach on visual servoing metrics, specifically on error in visual features (image area and image centroid), camera pose error, velocity profile and camera trajectory for PBVS, reverse-PBVS and part-IBVS. We consider a single test case and plot evolution of these parameters over the entire run. The images captured by the MAV for this run are presented in figure 5 column 1. It can be seen from figure 6(a-e) that all the errors and velocity profile exponentially decay for PBVS, part-IBVS and reverse-PBVS. Since the PBVS employs the MAV's odometry, which is accurate because of simulation environment, therefore the camera pose error and velocity profile are smooth for PBVS as shown in figure 6(a,d). However, every part-IBVS iteration requires the computation of image moments which in turn are estimated from the segmentation network, thus the pose error, image moments error and velocity profile are sensitive to even small errors in segmentation, therefore their plots are jittery as shown in figure 6(c,e). Furthermore, the camera trajectory shown

Final Area (A) of parts Without Bayesian refinement								
Car Name	RB-wheel	RF-wheel	R-mirror	R-head	L-head	L-mirror	LF-wheel	LB-wheel
Car 1	9.3	7.3	1.7	4.9	1.2	-	8.7	7.8
Car 2	10.9	10.6	-	1.1	1.7	-	10.3	10.2
Car 3	9.2	9.9	2.2	2.3	-	1.5	9.2	10.1
Car 4	9.9	10.9	2.1	1.7	1.5	3.1	10.3	11.4
Car 5	8	16	2	-	-	-	7.8	7.9
Car 6	10.6	10.3	2.7	7.5	2.7	-	9.1	9.7
Car 7	11.8	-	2.3	-	-	1.7	-	12.4
Car 8	11.1	8.9	-	2	2.1	-	9.9	8.8
Car 9	10.7	10.6	-	2	2.6	-	10.1	10.1
Car 10	9.1	9.8	0.6	5.2	2	-	9.8	9.1
Car 11	9.8	9.8	-	-	-	4	10.6	10.2
Car 12	10.1	10.4	-	2.2	2.5	1.3	10	10.1
Mean area	10.0	10.4	3.2	2.9	2.0	2.3	9.6	9.8
Desired area (A^*)	10	10	2	2	2	2	10	10
Final Area (A) of parts With Bayesian refinement								
Car 1	9.8	8.5	1.8	2.9	1.5	1	9.2	7.5
Car 2	10.4	10.2	2.5	1.2	1.3	-	10.9	10.6
Car 3	9.7	10.1	2.2	2.8	1.7	1.4	10	9.6
Car 4	9.9	10.6	3.3	2.1	3.4	-	10.3	12.2
Car 5	10.2	6.2	2	2.2	2.5	-	10.7	-
Car 6	10.3	13.4	1.4	1.9	2.5	-	10.3	10.7
Car 7	10.3	11.8	1	-	10.3	1.9	9.7	12.7
Car 8	10.6	9.6	2.1	2.4	2.3	2.2	10.1	11.4
Car 9	10.9	10.5	-	0.8	0.8	-	10.5	10.5
Car 10	9.2	10.8	1.8	2.9	7.5	6.1	10.7	9.9
Car 11	9.8	10.4	-	3.1	2.9	-	10.9	10.5
Car 12	10.4	10.5	-	2.3	2	1.6	10.2	12.9
Mean area	10.1	10.2	2.0	2.1	2.1	2.4	10.3	10.8
Desired area (A^*)	10	10	2	2	2	2	10	10

Table 1: **Quantitative results for complete autonomous part inspection pipeline with and without Bayesian fusion in Gazebo simulation.** Here we compare the performance of our approach on 12 cars by measuring the final area (A) of a part captured by MAV in percent of the image area as compared with the desired area (A^*) for a part. The A^* (10% for wheels, 2% lights and mirrors) is a tuning parameter that decides zoom level for images. It can be seen that by using the Bayesian fusion both the failure cases reduce as well as the resultant segmentation error abs($A - A^*$) decrease i.e. the final area is closer to the desired area. The failure cases where the MAV is not correctly aligned with the vehicle are reported as '-'. The areas reported in the above table are computed using the ground truth labels of the corresponding parts in the resultant image.

in 6(f) is also continuous and it can be seen that MAV starts from a random pose and visits all the essential parts of the vehicle. Note that the top view of a different car shown in the same figure, is simply for a better visualization. It could also be seen from the same figure that the PBVS results in an incorrect pose while navigating the MAV towards the front, however the IBVS refinement significantly improves the performance of PBVS by navigating the MAV to the correct frontal pose.

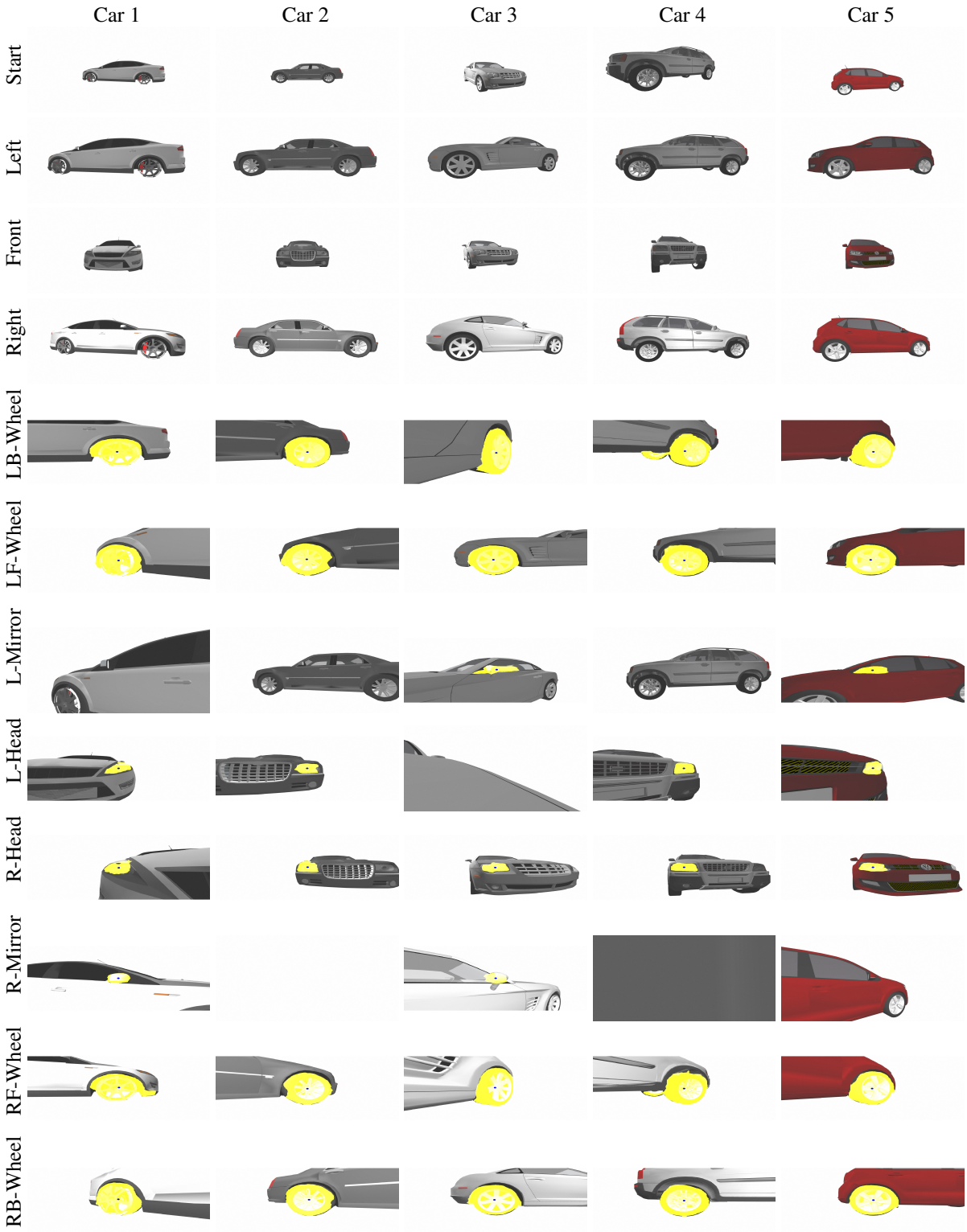

Figure 4: **Qualitative Results for entire part inspection pipeline.** This figure showcases the results of entire pipeline for five cars. The first row shows the images captured from a random initial pose. The images captured by the MAV from $\mathcal{F}_s^*$, for all sides are shown in second, third and fourth row. The subsequent rows show the images captured for every part and the corresponding part segmentation masks predicted by our network. Note that despite of the different shapes of cars and starting MAV at random poses, our approach aligns the MAV for visual inspection. The figure also shows some failure cases of our approach where the parts are out of MAV's field of view (for example, right mirror of car 2 and 4) or segmentation mask is not correct (left mirror of car 1 and 4).

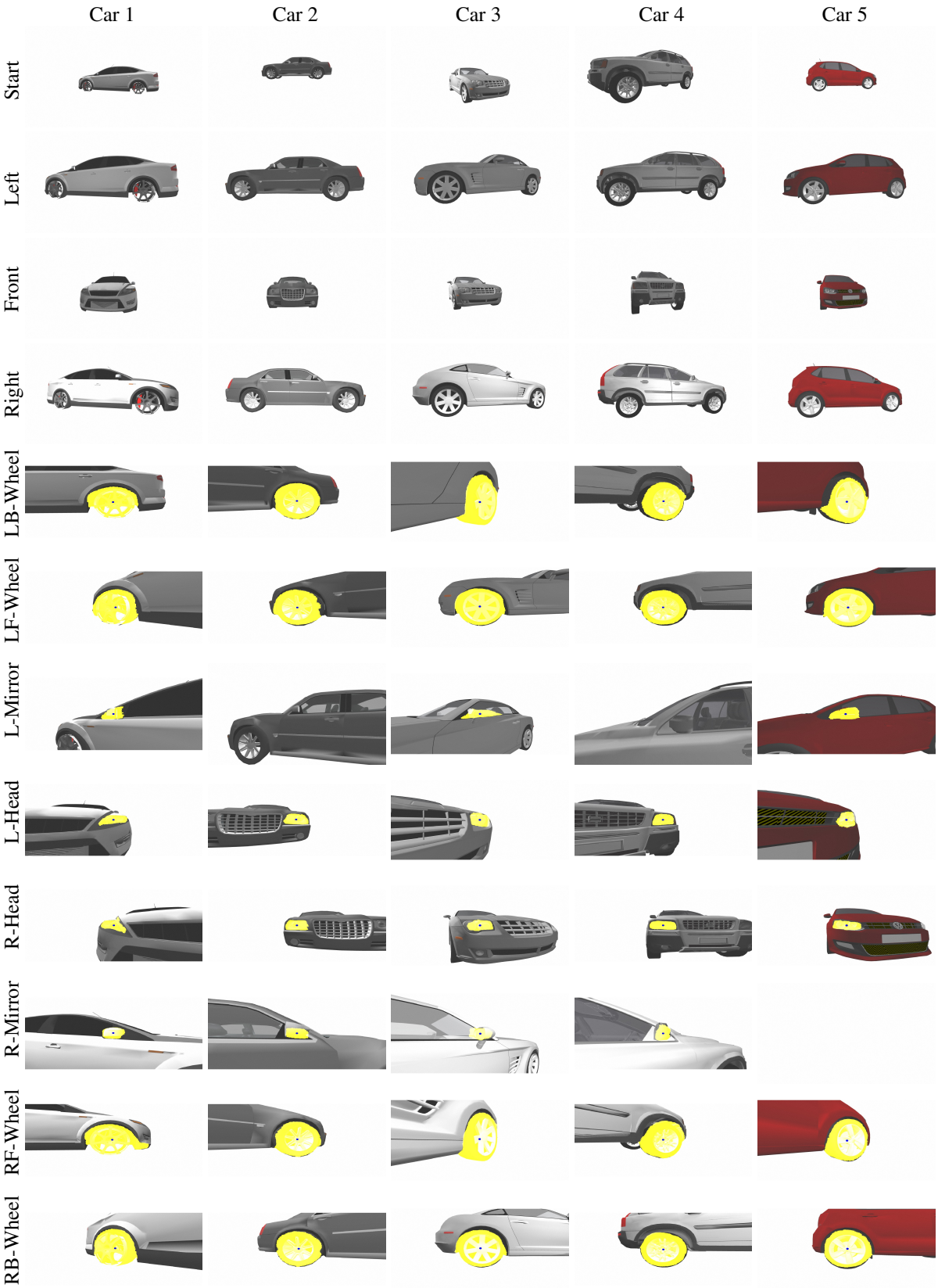

Figure 5: **Qualitative results for entire part inspection pipeline with Bayesian fusion in Gazebo simulation.** The failure cases are reduced to 3 because of the mulit-view Bayesian fusion of segmentation masks.

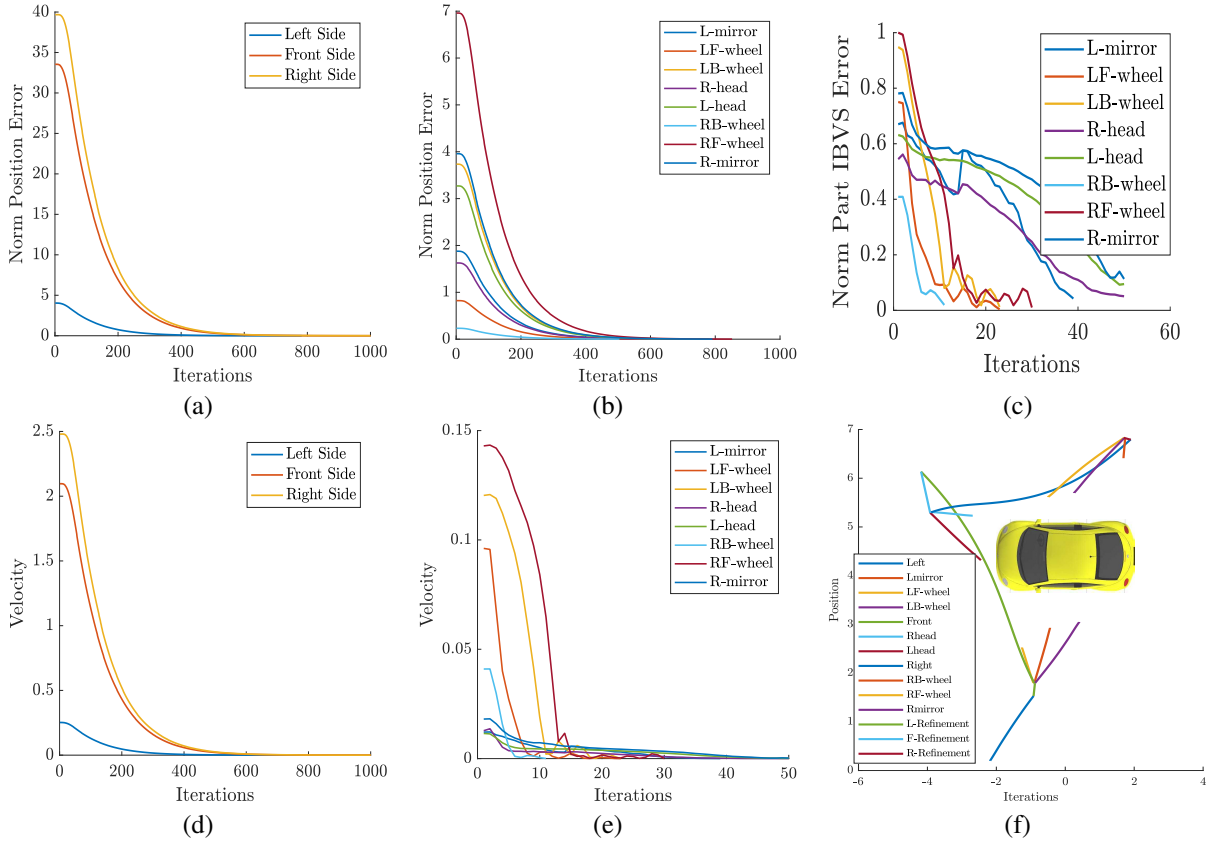

Figure 6: **Performance of visual servoing for a single car.** The performance of both PBVS and Part-IBVS on visual servoing measures are presented here. (a) Shows the error in camera pose ($\|\mathcal{F}_c - \mathcal{F}_s^*\|$) representing translation and yaw for the PBVS. (b) Shows the camera transformation error ($\|\mathcal{F}_c - \mathcal{F}_i^*\|$) for reverse-PBVS to every part. (c) Presents the error in visual features (image moments for the part-segmentation mask) over time for every part using Part-IBVS. The norm error between the current and the desired area of part-segmentation $abs(A - A^*)$ in percentage of area of image $(M \times N)$ and centroids current and desired segmentation mask in pixel $\|[x_g - x_g^*, y_g - y_g^*, A - A^*]\|$ is plotted. (d,e) Show the norm of camera velocities for PBVS and Part-IBVS respectively. (f) Shows the camera trajectory for inspection to every part combining our hierarchical PBVS and IBVS approach. The improvement due to IBVS refinement could be clearly seen in (f). Note that the feature error and velocity profiles are smooth and exponentially decay as desired.