Chapter 5

Conclusions and Future Work

The Middle Size league of RoboCup was the underlying scenario for this thesis: Fully autonomous robots, perceiving their environment with an omni-directional video camera were built and a computer vision system able to reliably localize the robot in real-time within this competitive environment was developed. All methods proposed in this thesis were applied at the world-championships in Lisbon 2004, and were demonstrated to work, while staying both efficient and robust. All the software, including computer vision, motor control, communication, and behavior runs on a single Pentium III 900 MHz processor. Region tracking is performed with 30 frames per second and feature recognition with 10 frames per second. This lower rate is not due to speed limitations, but solely because a higher rate is not necessary. Despite these high demands, the processor load is below 80 percent.

To the knowledge of the author, the developed system is the first vision system able to perform real-time recognition of a whole palette of shape-based features and uses them for robust navigation. This happens in real-time within a competitive and highly dynamic environment where occlusion frequently occurs. The following sections try to put this work in a more global context and discuss and indicate directions for further research.

To identify required future research, we ask whether and how the developed vision system could be generalized. We ask for the changes that have to be done and what is missing, to turn the specific vision system into a general vision system. Here, the demand is high: With *general vision system*, a system is meant, that allows to do everything that a human can do by means of his visual sense: Skiing, driving a car at night, reading a book, doing the dishes, etc.

5.1 Considering the System Dynamics Approach

The system dynamics approach (SDA), which is due to Dickmanns [26] and Wünsche [86], is certainly an essential achievement. It has been applied in a variety of applications [8, 50, 59, 83, 90]. However, although the general framework is clear, many problems still have to be solved to produce a truly general vision system. These problems will be described in the following sections.

5.1.1 Automatic Modeling

The most challenging problem is that of automatic modeling. Although the SDA was applied in many different scenarios, the types of considered objects were always user-specified. For autonomous vehicles driving on a highway, the model of the highway and other vehicles was predefined [27]. Although these models typically have some adaptable parameters, the scope of flexibility is very limited. In our RoboCup application, the model of the field lines is also pre-given and static. In the real world, there are many situations where unknown objects might appear. For instance, when a human hikes on a trail with stones and roots of trees sprawling on the way, these objects can have an unpredictable shape. A general system must be able to build models of these objects online. Of course, some kind of templates for the models could exist. For instance, stones might be approximated by polyhedrons; however the number of faces and their geometric constellation would have to be determined online by the system. Also, different objects might need different types of modeling. For instance, when trying to build a rescue robot, to extinguish fires, it is presumably inadequate to model the flames and fumes with polyhedrons. In contrast to rigid objects, their form changes continuously. When trying to build a framework that achieves automatic modeling, the focus should not be the creation of a large database of preexisting models. The complexity of a database would explode, since there is an infinite number of possible objects. It would be better to have a method that is able to efficiently create the models online from the observations as soon as they are needed. Of course, background knowledge in a database could assist this process.

A similar problem is currently investigated in the context of SLAM algorithms, which should allow localization and map building at the same time [84]. Here, parameters of objects are included in the system state and both the state values of the robot and the environment are estimated at the same time. However, typically only one type of object (i.e. simple artificial beacons) is considered.

Similarly, the motion model of objects is sometimes not known a priori. In RoboCup, for instance, different teams have robots with completely different motion properties. Some robots have omni-directional drives, others have a turnable front wheel with two rigid rear wheels, that are unable to move sidewards, for instance. Moreover, since the robots are autonomous and have a behavior that controls the movements, it is difficult to find an appropriate motion model. In the future, one could try to use several hypothetical motion models at the same time and choose the most likely one after having observed the behavior of the object for a while. This idea is similar to that of Monte Carlo Localization where several hypotheses for the robot's position are considered, until one main cluster crystallizes.

5.1.2 Automatic Learning of Feature Detectors

All applications where the system dynamics approach has been applied, use a set of predefined feature detectors. However, when the problem of automatic modeling is solved, and the type and appearance of objects are very flexible, the appearance of features can greatly vary. Thus, methods able to create appropriate detectors online have to be found. When a new type of objects comes into sight, even new types of features might be "invented", depending on the properties of the object.

5.1.3 The Problem of Feature Selection

Although Wünsche described a method of how to select appropriate features in [86], the underlying environment in which he developed the approach was very simple. A few polygonal, white objects with a clear black background were considered, and the visibility test of a feature consisted only of a test measuring whether the corresponding face of the feature was directed away from the observer. The problem is, that if the scenario and the internal model of the environment is complex - for instance a 3D model with some hundreds of faces - the visibility test becomes computationally expensive. Furthermore, not only the visibility, but also whether and how a feature can be detected in an image has to be calculated. Even if the visibility of a feature has been determined, the kind of detector that is suitable must be calculated. If the background has the same color as the object, it might not even be possible to detect it. One idea to test the visibility and to efficiently determine the appearance of a feature would be to use 3D rendering hardware to project the internal estimate of the model onto an artificial image plane. Then one could use the rendered image to determine whether a feature is visible or not, and how it appears.

Another interesting approach would be to investigate how far the emerging research field of *kinetic data structures* [36] could be used to develop efficient methods for the visibility test. Here, the underlying idea is to develop data structures that are able to efficiently adapt to changing situations without the need for consecutively computing the entire solution from scratch.

5.2 Top-Down Versus Bottom-Up Methods

There are two principles that have to be combined in a vision system: On one hand, there are bottom-up methods that take solely the video stream as input, try to detect some features (i.e applying a Canny edge detector) and then try to group the features and recognize objects or extract other information like optical flow or disparity maps for stereo vision. In the extreme, these methods do not use prior knowledge about the domain of application.

On the other hand, there are methods like the system dynamics approach, which uses as much knowledge as possible to interpret visual information. They are much more efficient than bottom-up methods; however, their problem is that they need an initial estimate. Moreover, when run isolated, they are not able to recover from catastrophic errors. Thus, both approaches must be combined: The bottom-up method creates hypotheses for features and objects, which can then in turn be tracked using top-down methods.

The proposed region tracking method contains elements of both principles: On one hand, the method uses no prior user-specified information about the domain (except the colors). Thus it is able to track the green regions and extract the field lines, without having an initial estimate of the robot's pose. On the other hand, the method uses prior knowledge - it takes the results of the last frame and uses them as a starting point for the next. However, this knowledge is created by the method itself and not super-imposed a priori. Having extracted the field lines, the feature recognition method also works without prior knowledge of the robot's pose. It is able to reliably yield an estimate of the system state. Once an estimate is available, tracking and updating cannot be done more efficiently than using the system dynamics approach. Having ended in this top-down recursive estimation phase the low-level bottom-up processes still have to run, since new objects could enter the field of sight.

Interestingly, the duality between bottom-up and top-down methods can be observed in some modern art paintings. For instance, in Dietrich Stalmann's painting in figure 5.1, a face is shown, which is over-painted in a second layer with colored, artificial strokes. What's interesting is that the human visual system is able to perceive and operate with these strokes, although they can certainly not be attached to an existing model of an object. Their form and constellation is new. Maybe this is the specific attraction of this painting, that the background shows an object that is well-known in its type, but that at the same time a different layer exists, which shows physically not interpretable strokes. Although a painting is a static image, these two layers can be seen as representative of top-down methods like the system dynamics approach, and bottom-up methods, which are required to initially detect and learn the shape and appearance of new objects and to create initial hypotheses for the top-down methods.

5.3 Criticizing the Proposed Feature Recognition Approach

In this thesis a constructive feature recognition approach was proposed, which allows the real-time recognition of a whole palette of different features. Here, the term "constructive" should indicate the way in which recognition is performed. High-level features or objects, such as the center circle, are constructed from smaller parts like arcs for instance. Although the method has proved to be robust and efficient and although it improved the robustness of the overall localization significantly, it can be criticized:

Although the method uses prior information of the task domain, it does not exploit the knowledge in a degree it could be done. The reason is, that the method does not make a hypothesis for the robot's position, but rather tries to construct features like the center circle without using this information.

Significant improvements of efficiency could be made, if hypotheses about the robot's position would be made at a very early processing step. Then, the knowledge about the task domain could be exploited in a higher degree. During the initial phase, or when recovering from erroneous localization, several hypotheses (i.e 6-12) about the robot's position could be considered in parallel, and feature recognition could be performed, with each recognition process being able to fully exploit the available prior information.

Another problem is, that the proposed feature recognition approach is very specialized to the geometry of the RoboCup field lines. Furthermore, the prior knowledge, that is used during recognition, is implicitly integrated in the source code. It is of utmost important to develop methods, that do not integrate processing and prior knowledge in such an inflexible way. It would be nice to have a method that would take a model of the environment as input and that would automatically organize and maybe even program the feature recognition in a way that is most efficient for the given domain.

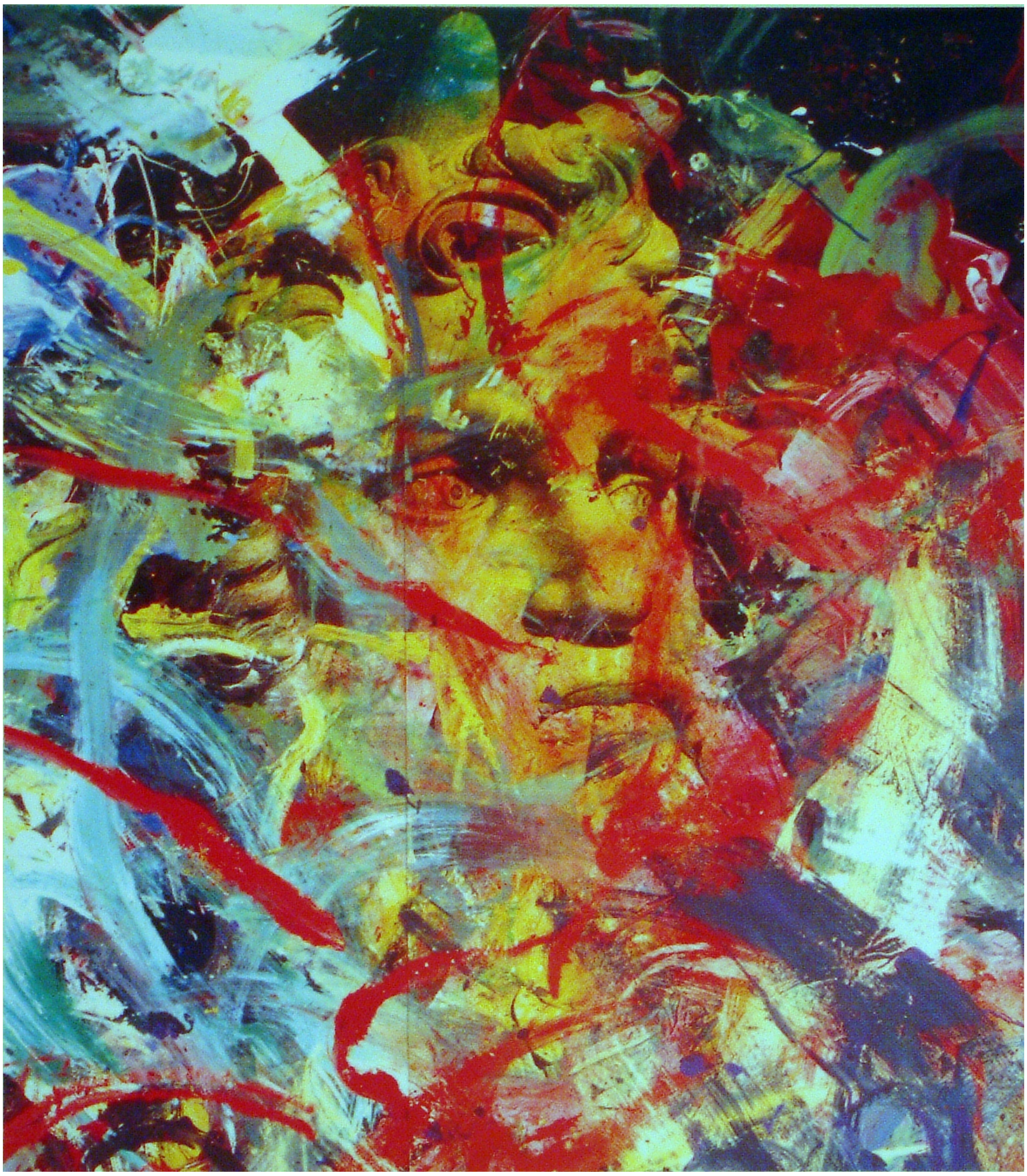

Figure 5.1: This painting from Dietrich Stalmann consists of two different layers. The background layer shows a realistic human face. The foreground layer consists of colored strokes, whose interpretation is difficult for a visual system. No physical objects exist that would give rise to such a constellation of strokes. However, the human visual system can detect them and operate with them, although they are completely new for the system. Thus, the two layers can be seen as representative of top-down methods like the system dynamics approach, which could be applied for tracking the face and bottom-up methods, which are able to perceive and learn unknown shapes an yield first hypotheses for objects.

5.4 The Problem of Light

Three influences determine the appearance of the color of an object in the image: The object itself, the illuminating light and the receptors detecting the reflected light. Thus with only a single pixel in an image, it is not known how much of its color is caused by the settings of the camera (hue, gamma, saturation, exposure,...), by the object's surface, or by the light. A yellow sheet of paper, for instance, might be a white sheet of paper lightened by a yellow lamp or a yellow paper in daylight.

The overall physics involved in this process are complex: A light source emits electromagnetic radiation at different frequencies with different energy. The spectrum of emitted wavelengths varies with the angle of the rays leaving the light source. The radiation is then reflected, refracted, absorbed, sometimes even re-emitted at different frequencies and these properties again depend on the object's surface properties and on the angle of both the incoming light and the angle at which the object is viewed. The light enters the optical system of the camera where lenses produce chromatic aberrations and other effects. Then it reaches the receptors, which have specific spectral characteristics. Their responses depend on the power and frequencies of the stimulating light. Effects like oversaturation may appear. And finally, not a single object and light source is present, but rather a whole set of objects and light sources, including ambient light. Light reflected by one object can still reach another object before it enters the camera.

The image in figure 5.2 is an example of how complex situations can be. Assume a vision system which should enable a robot to grasp one of the depicted objects in the showcase and that the vision system is based on the system dynamics approach. According to the philosophy of the approach, not only a model of the objects, but also of the light sources has to be used. However, with the demand of flexibility, the system must be able to create the model by itself. That is, it has to infer the position, number and kind of light sources and the surface properties of the objects. Here, the problem is that if the model is not detailed enough, a lot of effects that are present in the image cannot be explained. On the other hand, if the model is precise, the number of parameters that have to be estimated will be huge.

For instance, let us consider a physically inspired analytic reflectance model like that of [20]. Here, the model considers how the light is reflected from a single point on a surface, depending on the angle from which the point is illuminated and the angle from which it

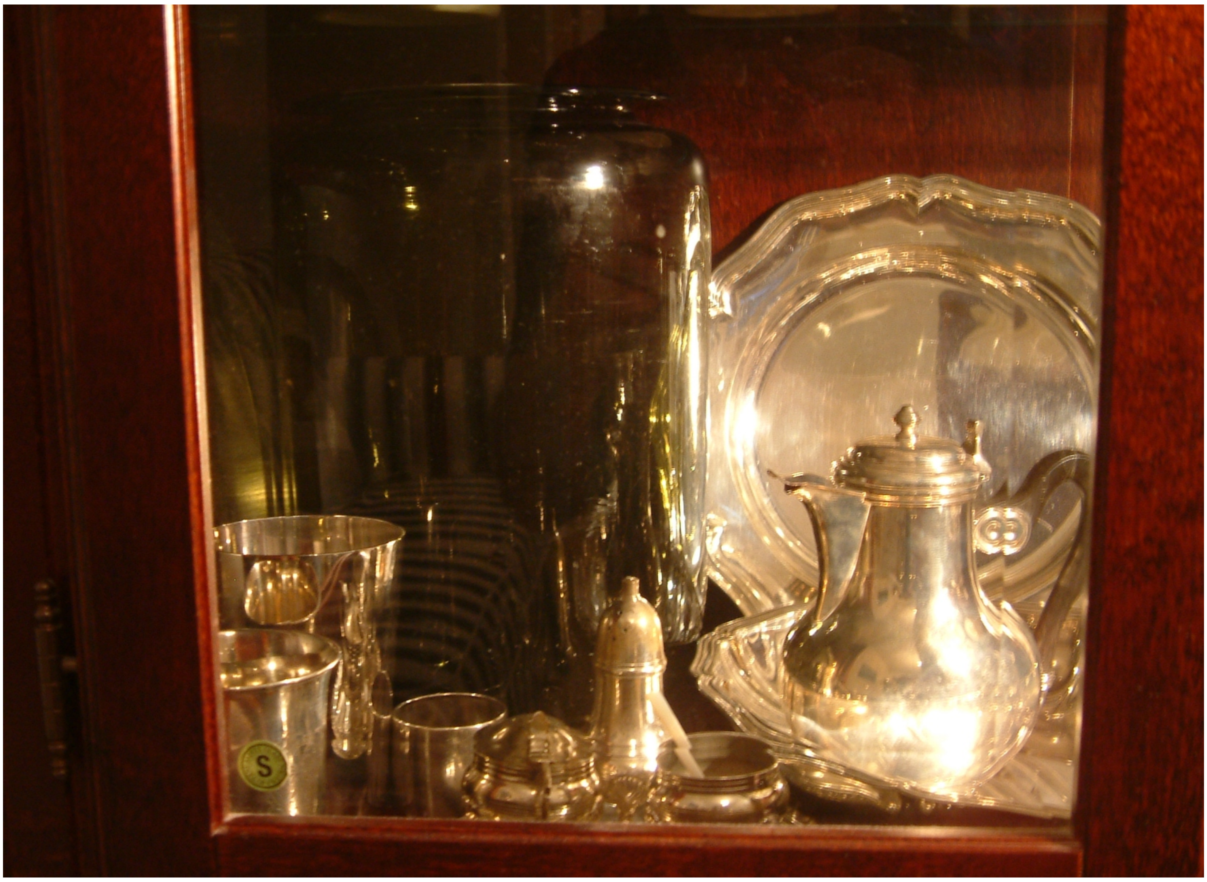

Figure 5.2: Silver objects and a glass vase are viewed through the glass door of a show-case. The striped cover of a sofa in the surroundings of the case is reflected within the glass, its colors blending with colors of the interior objects. At the same time, the silver objects reflect their surroundings, which in total, produces a rich spectrum of visual effects, which are difficult to model and predict.

is viewed. Without going into the details of the model, we look at a summary of the used symbols in table 5.1. As can be seen, a lot of parameters have to be specified, in order to model the reflectance properties of a single point. Moreover, the considered model is just an approximation of the reflectance of real materials, and it describes only a subclass of materials. That is, different models would have to be applied for different materials, which first requires material identification. But even if modeling of the light sources and the surface properties can be achieved, it becomes difficult to predict the appearance of features in real-time. Assume that we want to track the silver cup at the bottom left of figure 5.2, and that we have determined a set of visible features, one of which is the right

α	Angle between $\mathbf{N}$ and $\mathbf{H}$	θ	Angle between $\mathbf{L}$ and $\mathbf{H}$ or $\mathbf{V}$ and $\mathbf{H}$
λ	Wavelength	D	Facet slope distribution function
d	Fraction of reflectance that is diffuse	$d\omega_i$	Solid angle of a beam of incident light
E_i	Energy of the incident light	F	Reflectance of a perfectly smooth surface
f	Unblocked fraction of the hemisphere	G	Geometrical attenuation factor
$\mathbf{H}$	Unit angular bisector of $\mathbf{V}$ and $\mathbf{L}$	I_i	Average intensity of the incident light
I_{ia}	Intensity of the incident ambient light	I_r	Intensity of the reflected light
I_{ra}	Intensity of the reflected ambient light	k	Extinction coefficient
$\mathbf{L}$	Unit vector in the direction of the light	m	Root mean square slope of facets
$\mathbf{N}$	Unit surface normal	n	Index of refraction
R_a	Ambient light reflectance	R	Total bidirectional reflectance
R_d	Diffuse bidirectional reflectance	R_s	Specular bidirectional reflectance
s	Fraction of reflectance that is specular	$\mathbf{V}$	Unit vector in direction of the viewer
w	Relative weight of a facet slope		

Table 5.1: This table summarizes the symbols used in the physically-inspired reflectance model [20]. The length of the list indicates that a huge amount of parameters have to be determined to model the interaction of light and objects in a physically-inspired way.

edge of the cup. To apply an appropriate detector, we have to predict the appearance of the feature in the image. However, since the material is reflective, the appearance depends on what is reflected, that is, on other objects and light sources. Figure 5.3 shows a cut out of the right side of the cup. As can be seen, the situation is even more complex, since the cup is viewed through a glass in the door, which itself reflects objects in the surroundings of the show case. Horizontal stripes of a sofa blend over the side of the cup. Only a weak edge arising from the side of the cup can be seen. To predict this complex appearance, even modern rendering hardware would be too slow, since ray tracing methods would have to be applied. Finally, even if modeling and prediction of the appearance could be accomplished in real-time, it is doubtful if the required precision could be achieved.

Of course, these considerations do not mean that the system dynamics approach could not be applied, but the question is how things should be modeled. Instead of using a physically deep model of the light and surface reflectance properties, one could try to use a model that is not physically inspired but can explain the appearance in a simpler way. For instance, one could model the surface of objects by several semi-transparent layers: The bottom layer for the intrinsic color or texture of an object, one layer for shading and illumination effects, one for specular effects, one for reflection etc. In a dynamic environment, specular highlights or shadows can move, even on a non-moving object.

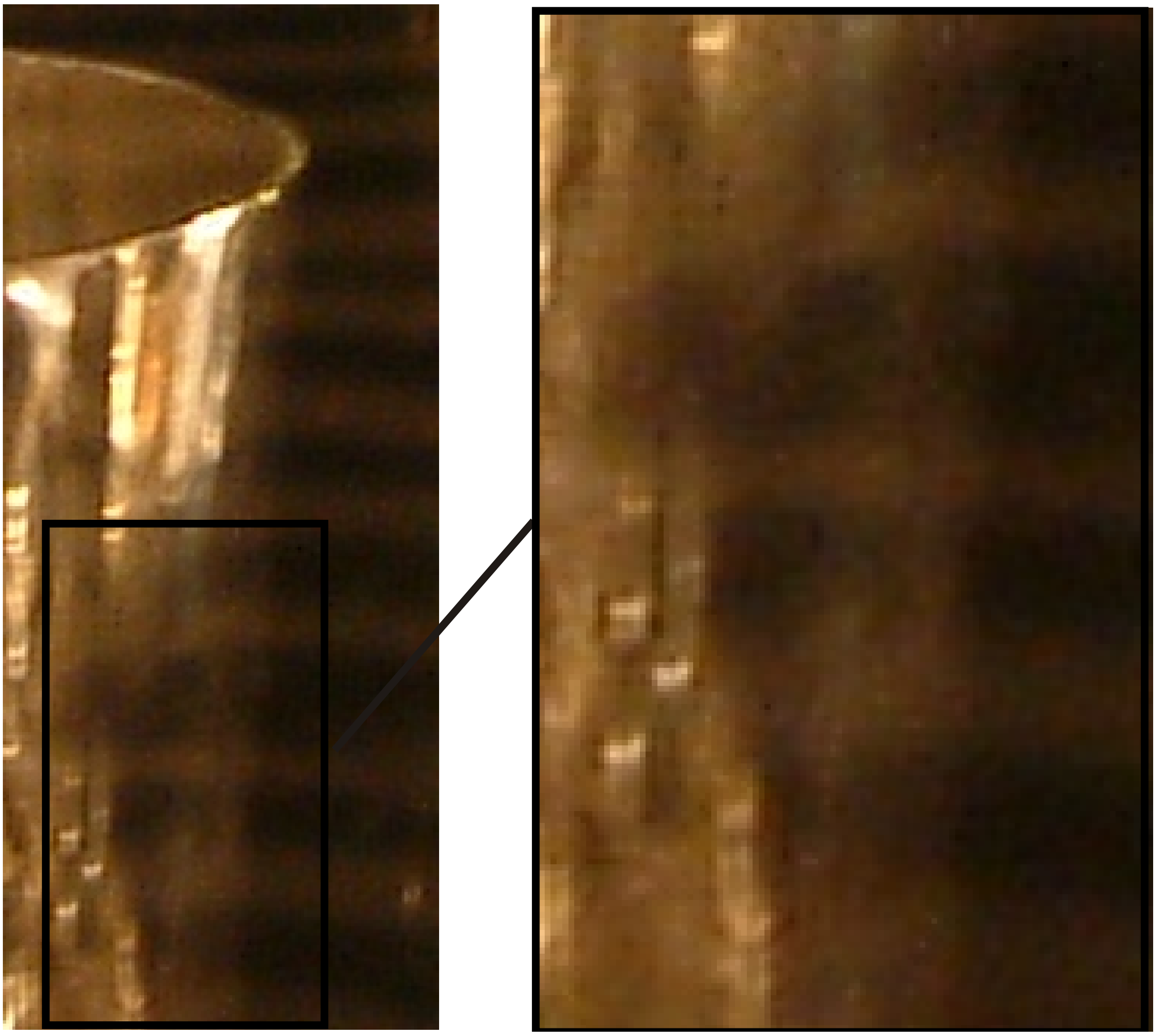

Figure 5.3: The left image is a cut-out of figure 5.2, showing a silver cup within a showcase. Assume that the cup is tracked with the system dynamics approach and that a detector at the right side of the cup has been determined to be visible. A cut-out of this right side is shown at the right part of the figure. To select an appropriate detector, the appearance of the expected edge of the cup has to be predicted. However, this prediction will be very complicated, since the cup is viewed through a glass door, which itself reflects objects in the surroundings. At the right side of the cup, horizontal strips can be seen that belong to a sofa located nearly.

However, instead of trying to explain these movements by other light sources and objects, one could try to create geometric and dynamic models of moving shadows and lights within the layers of the object's surface. Once this is accomplished, it is still possible to apply deeper models, and the inference of the corresponding parameters will probably be easier after having first derived the parameters of the simpler model.