Model-based recognition of 2D objects under perspective distortion

Stefan Wirtz* and Dietrich Paulus†

Institute for Computational Visualistics,

University of Koblenz-Landau, Universitätsstr. 1

D-56070 Koblenz

Kerstin Falkowski‡

Institute for Software Technology ,

University of Koblenz-Landau, Universitätsstr. 1

D-56070 Koblenz

(Dated: April 12, 2011)

We report on a case study showing on recognition of objects under perspective distortion in projected 2d images. We use symbolic descriptions and yield similar results as heuristic or statistical methods. The knowledge is modelled in so-called TGraphs which are typed, attributed, and ordered directed graphs. We combine the search in the state space with a maximum weight bipartite graph-matching and in consequence we reduce the numerous amount of hypotheses. Furthermore we use hash tables to increase the runtime efficiency. As a result we reduce the runtime up to a factor of five in comparison to the system without hash tables and achieve a detection rate of 90.6% for a data set containing 968 perspective images of poker cards and domino tiles. Therefore, we show that model-based object recognition using symbolic descriptions is on a competitive basis.

Keywords: Object recognition, Object detection, Knowledge-based recognition, Model-based recognition

I. INTRODUCTION

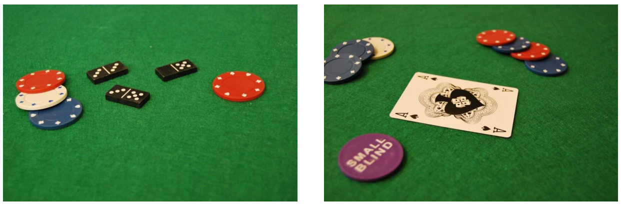
While heuristic approaches are commonly used to solve specific object recognition tasks, model-based approaches grant the opportunity for more general solutions. Furthermore, object detection using *symbolic descriptions* for model-based image analysis is still a challenge, although there has been research in this area since many years.

The advantage of symbolic descriptions is that usually no large amount of training data is required and man-made objects can often be described by symbolic descriptions as they are naturally provided by human experts. Nevertheless, it needs a lot of work to create a model of a complex object (e.g. of a building).

*Electronic address: wirtzstefan@uni-koblenz.de

†Electronic address: paulus@uni-koblenz.de

‡Electronic address: falke@uni-koblenz.de



(a)Domino tiles (b)Poker card

FIG. 1: Applications: recognition of domino tiles (a) and poker cards (b).

Indeed Google Earth provides lots of building-models which are designed in *COLLAborative Design Activity* (Collada), *Keyhole Markup Language* (KML) [1] and *Geography Markup Language* (GML) [2]. These models deliver symbolic descriptions of buildings which may give new relevance to model-based image analysis, because plenty of models are available.

One archetype of a system, that uses symbolic descriptions is ERNEST [NSSK90], which is a pattern recognition system developed in 1980 employing semantic networks in several application domains, like building recognition [Qui95] and speech understanding [NSSK90].

In this paper we present a case study showing the recognition of 2D objects in perspective images with a *task-independent pattern recognition approach*; we exemplify this on the recognition of domino tiles and poker cards (see FIG. 1). In a prospective study we will apply these results to the more complex 3D application area of building recognition. As in [WHP10], we use TGraphs [ERW08] as declarative knowledge description. We store the knowledge in schemata and instantiate concrete models using this knowledge. To find features in images we exploit the explicit knowledge given by the models. Therefore, we also use a *task-independent activity-control*, which manages the application flow of the system.

Nevertheless we extend the activity control to work with *subgraph partition* and an *efficient cashing* to increase the performance of the system. Furthermore the activity-control employs the Hungarian-Method, a fast graph- matching algorithm, whenever it is possible to reduce the assignment-task to a maximum weight bipartite graph-matching.

We introduce the related work in Section II. Section III describes our approach in detail. There we give the exact definition of the task and describing our models as well as the designed belief functions. We show and discuss our experiments and results in Section IV. A summary and ideas for future work can be found in Section V.

II. RELATED WORK

In principle, the object detection task for domino tiles and poker cards is simple.

The recognition of domino tiles has been used as a case study before, e.g. in [BHJ$^+$99]. A heuristic strategy is to detect circles and rectangles and count the circles in each half of the domino tile. Nevertheless, the detection of a rectangle in a perspective image, especial of a 3D object like a domino tile, is not trivial: one has to search for quadrangles and calculate the homography [HZ03] for them. Another approach is to apply *template matching* [TK09], which is widely used in face recognition. Bollmann et. al. [BHJ+99] use template matching to identify domino tiles. The advantage of template matching is that it delivers a similarity measure, but it is invariant neither to rotation nor to scale.

In the same way as domino tile recognition the poker card recognition can be solved; Also rectangles have to be detected and template matching can be used to classify the kind and number of the card. As an improvement one can use a *Optical Character Recognition* (OCR) system to identify the card characters.

Today, model-based approaches usually use statistics, appearance, shape or geometry, which mostly work without an explicit geometrical model schema [CSS06, SLS06]. In this work we only use models where the modelling of the objects is in an explicit form, like it is used in Computer Aided Design.

[SDP09] presents an approach to model 3D objects with an attribute grammar formalism, where the focus of the paper lays on 3D buildings. One result was an XML schema for spatial attribute grammars. In [MAD09] knowledge-based image understanding is treated. The knowledge is coded declaratively in a production system. The authors in [HN08] use low-level image analysis with bottom-up processing for high-level interpretations, which were evaluated with the recognition of structures in building facades. Other systems exist which use ontologies such as OWL[3] and combine them with uncertain knowledge for finding concepts in a domain, e.g. in [Hoi10, RSH09].

Using models for explicit knowledge representation, there are two main strategies how a controlling algorithm can handle the analysis process (cf. [Nie90] p. 240ff). On the one hand, there is the *data-driven* strategy, where the segmentation objects found in the image serve as an initialisation for the analysis. Based on the segmentation objects, the best possible model is sought. On the other hand, the *model-driven* strategy works from model to the image data. Each model determines whether it is contained in the image or not and tries to locate its elements. With a data-driven strategy and a good segmentation, we quickly find the related model; with a model-driven strategy one searches selectively for segmentation objects.

We choose a *hybrid approach* where data-driven models become pre-evaluated and then we selectively search model-driven for segmentation objects.

Uncertainty theories, such as Bayes [TK09], Dempster-Shafer [BKI06] or fuzzy sets [BKI06] define how to deal with uncertain, insecure or vague knowledge. They provide a representation formalism for uncertain information and reasoning strategies. The Dempster-Shafer as belief propagation is used by Hois in [Hoi07], where Neumann uses Bayesian compositional hierarchies for scene interpretation [Neu08].

The knowledge-based processing strategy was popular in several areas, also in the image analysis, in the 80s.

Semantic networks were introduced and successfully used for image and speech analysis [KSN92, Sag85, NSSK90]. The formalism ERNEST [NSSK90] combines representation of concepts in a graph, using sparse edge types, with a task-independent control of A*-basis. There the search for the best association can be considered as a path search which can also be performed if not all states are generated.

An overview of knowledge-based systems for object recognition is given in [CL97]. Such systems need to be robust to errors in segmentation data as well as predominantly invariant to changes in image acquisition.

III. APPROACH AND TECHNIQUES

The basis for this approach are the models which contains the declarative knowledge (Section III B) and procedural knowledge (Section III D). With these models the activity control (Section III E) starts to recognize poker cards and domino tiles. The hypotheses of found poker cards and domino tiles are evaluated for each assignment (Section III F) and the entire model. The runtime efficiency is optimized by vertex-, subgraph-, and method-caching (Section III H).

A. Problem statement

The aim of this case study is to show that an object recognition system which works independently of the application domain (see FIG. 2) is able to detect objects,

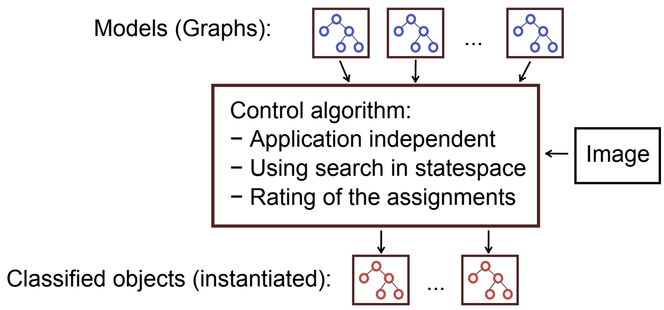


FIG. 2: Principle of the image analysis system.

given the symbolic description in a model. These models contain the declarative and procedural knowledge as well, which provides the procedures to detect model elements in the image and to rate the assignment of a model element to an image element. During the matching of elements found in the image and elements given by the model, a large number of hypotheses arise. Therefore, it is necessary to find an efficient way to deal with these hypotheses.

B. Declarative knowledge

The *declarative knowledge* for our model-based object recognition system is represented in particular models that conform to the *STOR reference schema* (section III B 1) enhanced with specific *object semantics* (section III B 2).

1. The STOR reference schema

The STOR reference schema is a *graph-based meta-model*, that represents comprehensive knowledge from the field of Computational Visualistics (CV) and enables (automatic) algorithmic processing of conforming models. It is structured in *five basic packages* comprising different aspects of Image Processing (IP), and Computer Graphics (CG), completed with a particular *semantics package*, that can be supplemented with sub packages for specific application areas.

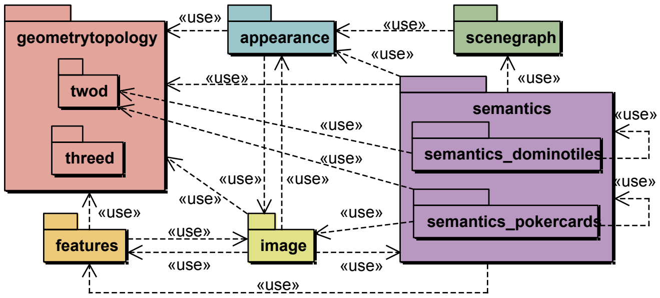


FIG. 3: STOR reference schema (packages).

Figure 3 gives a coarse-grained overview of the schema packages and their relations. The `image package` contains images and typical image parts used in IP and CG, like pixels, edges, and regions. It explicitly distinguishes between an image as a model part and an image as a physical object. The `feature package` comprises typical elements from IP, e.g. blobs, corners, and key points with corresponding descriptors, that are results of various feature detection and extraction algorithms. The `geometry/topology package` contains typical geometry and topology elements from CG, i.e. the structure of an object and the positions of its parts in a specific coordinate system. Geometric entities are points, line segments, different kinds of faces, and specific collections (e.g. triangle lists). Topological elements are relations between geometric entities to describe higher order substructures. There are sub packages for 2D and 3D geometry and geometric objects are firstly modelled generally and then specialized in 2D and 3D. The `appearance package` comprises object properties like colours or textures. Appearance entities usually correspond to geometric entities. An object can have more than one appearance for different circumstances. The `scene graph package` contains typical scene graph entities used in CG. A scene graph forms some kind of tree that describes global object positions in a specific terrain via transformations from different local coordinate systems.

The `semantics package` contains an abstract element, named `SemanticObject` that builds the bridge between semantic elements and all other packages. For concrete application areas semantic, sub packages may be plugged in to the schema. All semantic elements in any semantic sub package have to be derived from the semantic object.

The schema bases upon the *TGraph technology* [ERW08, Ebe08]. TGraphs are a powerful graph concept for the explicit representation and efficient handling of knowledge in terms of entities with attributes and relations to each other. In existing object recognition systems such a representation is calls "symbolic description". TGraph nodes and edges are typed and attributed first-class citizens, for which multiple inheritance is allowed. Edges are directed, and nodes, edges and incidences are ordered. When information is represented by TGraphs, entities are modelled by vertices, their occurrences are modelled by edges and sequences are expressed by incidences.

TGraphs conform to TGraph schemas, which can be specified in the language graph UML (grUML), a subset of Unified Modeling Language (UML) class diagrams with graph semantics. When a grUML diagram is transformed to a TGraph schema, classes are mapped to vertex types, associations are mapped to edge types, class and association attributes are mapped to vertex and edge attributes, specializations are interpreted as type inheritance and multiplicities are interpreted as degree restrictions.

TGraphs are implemented in the elaborated API Java Graph Laboratory (JGraLab), that is based on symmetric incidence lists. It allows the efficient creation, manipulation, and traversal of TGraphs and TGraph schemas according to the TGraph metaschema. Furthermore it provides various utilities, e.g. for the conversion of a given grUML diagram to a TGraph schema and the generation of a Java API layer according to this TGraph schema, where schema entities are mapped to native Java constructs.

Based on this technology, models conforming to the STOR reference schema can easily be subject to all kinds of algorithms that create, manipulate, enhance, and transform the contained data [FE09a].

The STOR reference schema is independent of concrete application areas and in an application not all packages or elements have to be used. In our model-based object recognition case study we use the packages image, geometry/topology and semantics (section III B 2).

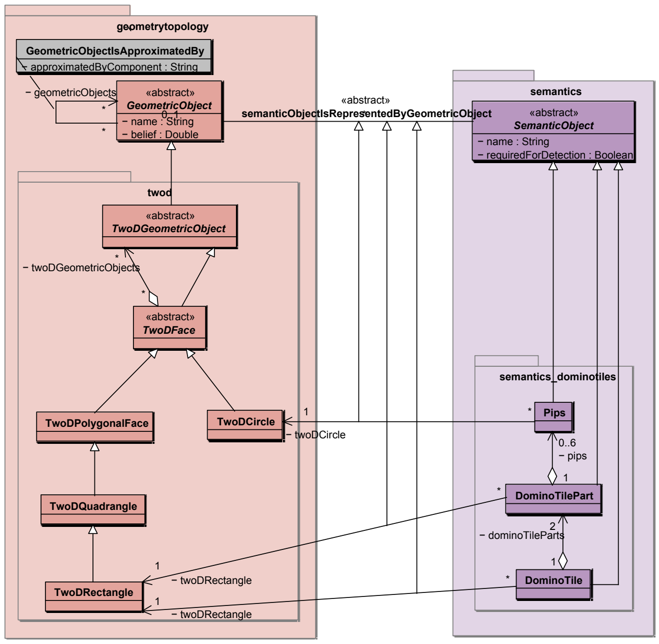


FIG. 4: Domino tile reference schema elements.

2. Object Semantics

We extended the STOR reference schema for poker card and domino tile recognition. Figures 4 and 5 represent those elements of the STOR reference schema that are currently used for model-based object recognition, including the semantics sub packages `semantics_poker cards` and `semantics_dominotiles`.

We use specific `representedBy-` and `consistsOf-` edges as well as the concepts `SemanticObject`, `GeometricObject` and `Region`.

Figure 4 shows the reference schema elements used to describe a domino tile. A `DominoTile` consists of two `DominoTileParts` and both concepts are represented by a `TwoDRectangle`. A `DominoTilePart` consists of zero to six `Pips` that are represented by `TwoDCircles`. The semantic parts of a domino tile are all specializations

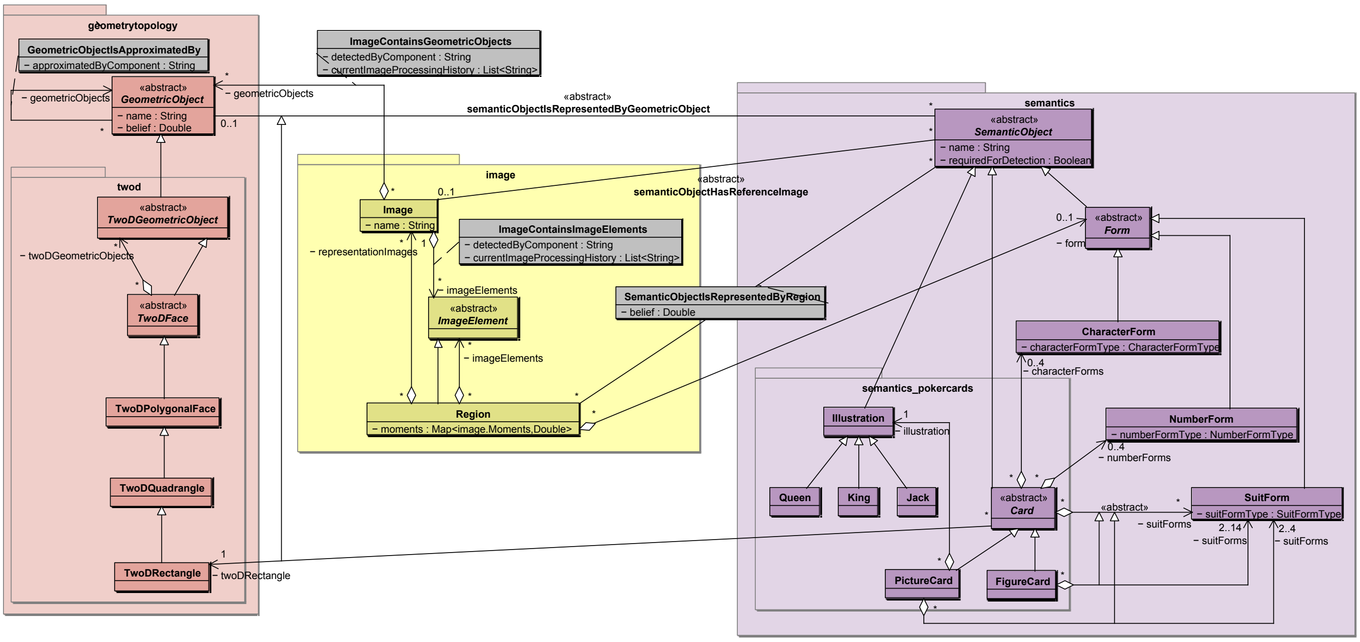


FIG. 5: Poker card reference schema elements.

of `SemanticObject`. `TwoDRectangles` and `TwoDCircles` are specializations of `TwoDGeometricObject`.

Figure 5 shows the reference schema elements used to describe a poker card. A `Card` can consist of `SuitForms` (Diamond, Heart, Spade, Clubs), `CharacterForms` (A,J,Q,K) and/or `NumberForms` $(0 \dots 9)$. The `Card` itself is specialized to `FigureCard` and `PictureCard`, so that the multiplicities of `SuitForm` can be differentiated. A card is represented by a `TwoDRectangle` and the `Forms`: `SuitForm`, `CharacterForm` and `NumberForm`. Forms are represented by a `Region`, which is characterized by its central- and Hu-Moments (included in the map `moments`) [Hu62]. The poker card semantics parts are also specializations of `SemanticObject`.

We describe domino tiles and poker cards in 2D although the objects are 3D, because both are classifiable based on their 2D front. That is the reason why a 2D modelling is in this case adequate. Anyway, with a given position estimation of the camera we generate a 2D model, which contains only the visible model elements from the estimated camera position and adapt the geometry according to this position.

With this enhanced reference schema, we construct reference models. In our case studies, we have 80 reference models of 28 domino tiles and 52 poker cards.

C. Procedural knowledge

The *procedural knowledge* was manually added to the generated schema API layer. The procedures are wrapped in components conforming to the *STOR component concept* [Fal10, FE09b], which supplies an experimental environment for CV to construct, analyze, and evaluate solutions. To find the occurrences of our model elements in the image we need methods for circle, rectangle, card-colour, and character detection (see section III D).

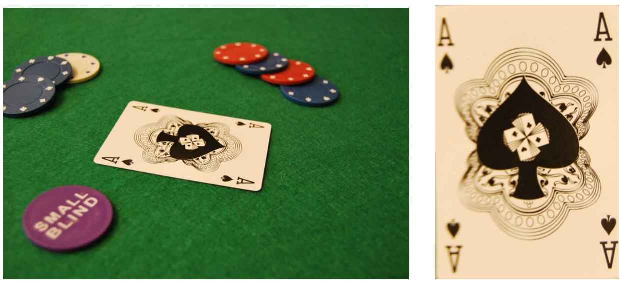
D. Segmentation

As described before we model next to declarative knowledge task-dependent procedural knowledge in our models. The main focus of this case study is to evaluate, if model-based object recognition with symbolic descriptions works properly. Admittedly the segmentation is necessary to fulfil this task. Hence we describe our methods in detail.

The *Circle detection* works with a modified Hough-Transformation circle detection operator from the PUMA library [4].

The *Rectangle detection* locates quadrangles in the image. Therefore, we segment the contours and approximate with them polygons. A quadrangle can be found with a convex polygon which consists of four corner points. Then we calculate the homography between the found quadrangles and the given model rectangles. The elementary functions for this method are taken form the OpenCV library [5] and wrapped to our component system with the JavaCV library [6]. The detection rate is about 100% in our case study, but it delivers a lot of false positives which have to be filtered out during the image analysis by the activity-control. Figure 6 shows a exemplary result of the rectangle detection.

As the *Card-colour* and *character detection* work equivalently, we explain only the card-colour detection in the following.



(a)Poker card image (b)Rectified poker card

FIG. 6: Example of an poker card image (a) and the detected and rectified quadrangle (b).

At first we search for potential regions of card-colours in the image. Therefore, we segement again the contours. Every contour is a potential region for a card-colour. For the region of the image where we located the contours we detect the colour of the contour. If the colour is red, the card-colour could be heart or diamond, if it is black, it could be spade or clubs and if it is neither black or red, it will be no card colour.

For the remaining regions, we calculate the Hu-Moments [Hu62] and the thinness ratio (equation 1):

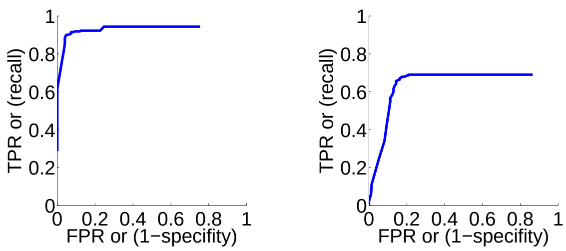
$$tr = \frac{4\pi \text{ area}}{\text{perimeter}^2} \ . \quad (1)$$

Then these features are compared to the training data of the card-colours. We only compare the features to the classes specified before (diamond and heart or spade and clubs); we use the Mahalanobis-distance:

$$d(x, \boldsymbol{\mu}_A) = \sqrt{(x - \boldsymbol{\mu}_A)^T S_A^{-1} (x - \boldsymbol{\mu}_A)} \quad (2)$$

where x is the feature vector, $\boldsymbol{\mu}_A$ the mean vector of the features of class $A \in \Omega = \{\text{Diamond}, \text{Heart}, \text{Spade}, \text{Clubs}\}$ and S_A is the covariance matrix of the features of class A. The region is classified to the class where the distance is minimal and does not exceed a given threshold.

The results for the card-colour detection are satisfying with a recall (true positive rate) of 93.2%, a precision (positive predictive value) of 96.1% and a specificity (true negative rate) 93.4%. These good results can also be seen in the *receiver operating characteristic* (ROC) - curve in figure 7(a). For ROC-curves it is valid that the greater the area under the ROC-curve is, the better is the result of the classification. A perfect classifiactaion has a value of one for the area under the curve. In the ROC-curve of figure 7(a) one can see that the recall never reaches 100%. The reason is that the segmentation of potential regions is not able to detect all regions, because especially in images with a strong perspective distortion, the regions are split into many small fragments.



(a)Card-colour ROC (b)Character ROC

FIG. 7: ROC-curves of the card-colour (a) and character detection (b).

The same problem is present in the detection of characters, but is much worse there, because characters have a capillary structure and oversegmentation will happen even more often than for the regions of card-colours. That is the reason why the rates with recall of 69%, precision of 72.3% and specificity of 78.6% are low for the character classification. The ROC-curve in figure 7(b) shows this deficit clearly. Here, the area is much less than the area for the card-colour detection and the recall never exceeds 70%. However, as we will show the model-based object recognition is able to compensate for these segmentation problems.

E. Pragmatic of the System

The approach created for this case study works similar to [WHP10]. A task-independent and A*-based activity control instantiate the models, fills the state space with hypotheses and controls the application flow. However, we extended this approach and combined the search in the state space with the Hungarian-Method [Kuh55]. This method is able to find the optimal assignment of model and image elements, if a cost or belief calculation for each potential assignment is possible. We use this method whenever the cardinality of a node is greater than one. This can be stated by four task-independent rules which are combined with graph search algorithms to handle the control problem.

Rule 1 *Check the multiplicity of the actual element for every type of the parts. If the multiplicity is equal to one use rule 1a with that part, otherwise use rule 1b.*

Rule 1 (a) *If a* `isRepresentedBy` *edge belongs to the given element, go to the element which is connected with a* `isRepresentedBy`*-edge and find the occurrence of the element in the image.*

Rule 1 (b) *If a* `isRepresentedBy` *edge belongs to the parts of the actual element, assign all parts of this type with the graph-matching algorithm: Hungarian-Method.*

Rule 2 *If no* `isRepresentedBy` *edge belongs to the actual element, expand the element, i.e., search for the part of this element which is not already processed. If all parts are already processed go to the parent element and repeat rule 2.*

Rule 3 *If the initialisation of an element was successful, limit the ROI of all parts which belong to this element.*

Rule 4 *If an element is not obligatory, try to find occurrences of the element in the image with rule 1. If it is not possible to find candidates for this element, the missing will not be punished.*

Figure 8 shows how the rules are included in the application flow.

Each step in the analysis is encapsulated in one state. The search for the best association can then be considered as a shortest path search, which can also be performed if not all states are generated. This search is controlled by the A^*-algorithm.

To be able to use the Hungarian-Method, each `SemanticObject` has to offer a judgement, according to

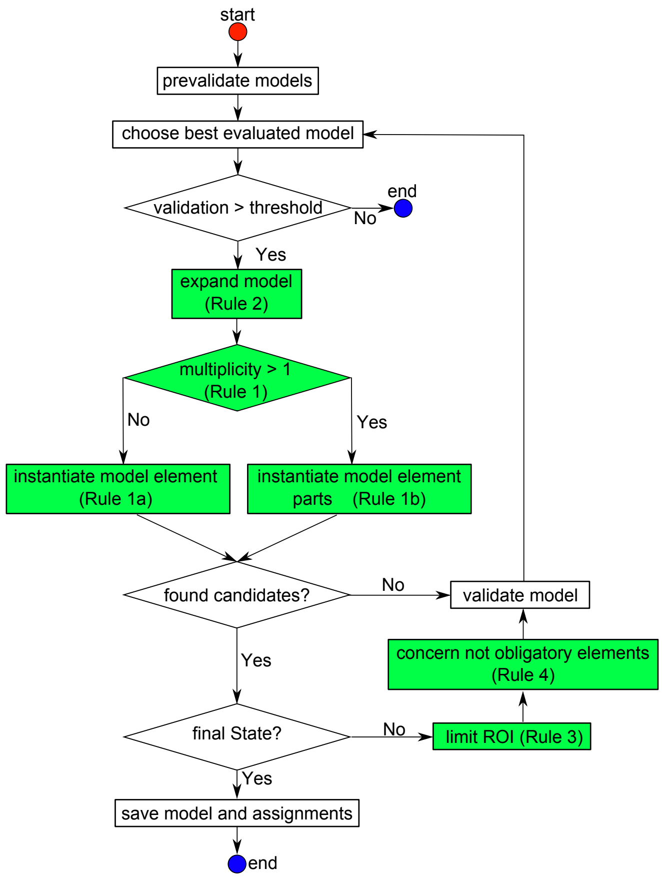


FIG. 8: Activity diagram of the activity-control.

the belief of Dempster-Shafer theory [Sha76], which describes how well an image element can be assigned to a model element and how missing assignments reduce the belief in the detection of this `SemanticObject`. These beliefs are combined as described in detail in section III F with the Dempster-Shafer combination rule.

F. Belief Functions

With the models and the activity control we assign image elements to model elements, but cannot say which model is plausible and which one is not. Therefore, we need belief-functions which provide qualitative values for each model.

In this case study we choose the Dempster-Shafer belief function, which allows to model the functions in a heuristic way without requiring the knowledge of the statistical distribution of our data. Dempster-Shafer provides also a convenient combination rule for independent beliefs. Therefore, we need the *basic probability assignment* (τ). A basic probability assignment must fulfil following conditions: Given the sample space $\Omega = \{a_k\}\ ,\ \ k = 1, \ldots, K,\ \ A \subseteq \Omega$, where a_k and A are arbitrary events.

$$\tau(\varnothing) = 0 \qquad (3) \qquad \sum_{A \subseteq \Omega} \tau(A) = 1 \qquad (4)$$

The combined τ with Dempster-Shafer is:

$$\tau_1 \oplus \tau_2 = \tau(A) = \begin{cases} 0 & : \quad A = \varnothing \\ \frac{\sum_{t \cap u = A} \tau_1(t)\tau_2(u)}{1 - \sum_{t \cap u \neq \varnothing} \tau_1(t)\tau_2(u)} & : \quad \varnothing \neq A \subseteq \Omega \end{cases} . \tag{5}$$

The Dempster-Shafer rule is commutative and associative. Accordingly, it is possible to combine various resources, but the resources have to be independent of each other because in general $\tau \oplus \tau \neq \tau$ is valid.

The belief in an event is then:

$$\mathrm{Bel}(A) = \sum_{A_j \subseteq A} \tau(A_j) \tag{6}$$

Using Dempster-Shafer we define three general functions which are valid for each `SemanticObject`. At first we need a function to combine the basic probabilistic assignments τ_i of two events of different sample spaces which assigns a belief to both, that an accordant element is found in the image (see eqn. 7). Therefore, the combination rule of DS (eqn. 5) is used like Quint made it in [Qui97]. Then the belief of a semantic object SO is

$$\begin{aligned} &\tau(\mathrm{SO} \mid \mathrm{E_{parts}}, \mathrm{E}_{rep}) = \\ &\quad \kappa_1\ \tau(\mathrm{SO} \mid \mathrm{E_{parts}}) \oplus \kappa_2\ \tau(\mathrm{SO} \mid \mathrm{E}_{rep})\ . \end{aligned} \tag{7}$$

With κ_1 and κ_2 you can weight the trust that this event really supports the belief in the `SemanticObject`, where $\kappa_1, \kappa_2 \in [0, 1]$. The information source E_{parts} describes the observation of the parts of a `SemanticObject` and the information source E_{rep} specifies the `GeometricObject` or `Region` which represents this `SemanticObject`.

We also need to know the belief of a `SemanticObject` given the information source E_{parts} (observed parts). Therefore, we combine all kinds of parts of this semantic object SO by

$$\begin{aligned} &\tau(\mathrm{SO} \mid \mathrm{E_{parts}}) = \\ &\kappa_1\ \tau(\mathrm{SO} \mid \{\mathrm{parts}\}_1) \oplus \kappa_2\ \tau(\mathrm{SO} \mid \{\mathrm{parts}\}_2) \oplus \\ &\qquad \ldots \oplus \kappa_N\ \tau(\mathrm{SO} \mid \{\mathrm{parts}\}_N)\ . \end{aligned} \tag{8}$$

Finally, we need the belief of a semantic object SO given its parts:

$$\tau(\mathrm{SO} \mid \{\mathrm{parts}\}) = \frac{1}{N} \sum_{i}^{N} \tau(\mathrm{SO} \mid \mathrm{parts}_i)\ . \tag{9}$$

These three functions are the same for each semantic object. They have to be supplemented by basic probabilistic assignments for the specific cases. Some important basic probability assignments will be presented in the next equations 10 - 12, but be aware that there are more specific belief functions.

The assignment of a semantic object to its representation in the image is most import for classifying a object. Therefore, we need a function to rate the assignment of a card-colour CO to a region Reg, where $(x_{\mathrm{M}}, y_{\mathrm{M}})$ and $(x_{\mathrm{S}}, y_{\mathrm{S}})$ are the centroids of the model region and the segmented region:

$$\tau(\mathrm{CO} \mid \mathrm{Reg}) = e^{-1a \sqrt{(x_{\mathrm{M}}-x_{\mathrm{S}})^2+(y_{\mathrm{M}}-y_{\mathrm{S}})^2}} \quad . \tag{10}$$

The factor a depends on the size of the rectified subimage. In our case the subimage is always 300×450 and the factor $a = 0.08$.

An equivalent function is needed in the domino case for the assignment of a pip $p \in \mathrm{P}$ to a circle $c \in \mathrm{C}$, where (c_x, c_y) and (p_x, p_y) are the centre points of the circles, and c_r and p_r are the radii. We choose the exponential form to prevent that circles are assigned to wrong and distant pips:

$$\begin{aligned} \tau(p \in \mathrm{P} \mid c \in \mathrm{C}) &= 101 - 200^q - 50 \mid c_r - p_r \mid, \\ q &= \sqrt{(p_x - c_x)^2 + (p_y - c_y)^2} \quad . \end{aligned} \tag{11}$$

We also need a belief function which penalizes missing assignments of segmentation objects and/or model elements in the following way:

$$\begin{aligned} \tau(\mathrm{dr}_i \mid \mathrm{P}) &= e^{-\frac{1}{2}x^2} \\ x &= \mid\mid \mathrm{C} \mid + \mid \{\text{given pips}\} \in \mathrm{P} \mid \\ &- \mid \{\text{associated pips}\} \in \mathrm{P} \mid\mid \quad . \end{aligned} \tag{12}$$

The choice of the functions (11) and (12) defines if it is better to drop an assignment or not. The more a missing element is penalized and the more $\tau(p \in \mathrm{P} \mid c \in \mathrm{C})$ forgives differences between pips and circles, the more segmentation objects are assigned.

With these functions we now have the ability to rate the assignments of model elements to segmentation objects. Furthermore, the combination of the functions gives us the total belief that a specific model was detected in the image. In this case study the basic probability assignments are equal to the Dempster-Shafer belief of equation 6, because all our basic probability assignments describe atomic events.

G. Belief Propagation

During the analysis process several hypotheses for models and assignments will emerge. Therefore, we perform a search in the state space, where we create a state for each hypothesis. As a result we need a way to compare hypotheses of different analysis steps beside the belief functions. We use the $\epsilon - A^*$ algorithm, with

$$f(x) = g(x) + \epsilon \ h(x) \quad . \tag{13}$$

In our case $g(x)$ equates the belief in a model and $h(x)$ equates the belief of a model, where all not yet assigned model elements are treated as perfect assignments with a belief of 100%.

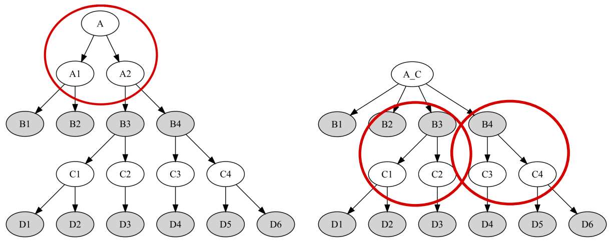
With the models, the task-independent activity-control, the belief functions, and a belief propagation we have all parts to find domino tiles and poker cards in images.

H. Caching

During the image analysis, many hypotheses arise, which means that we have to handle a lot of states in the state space. Furthermore, a lot of belief calculations for the individual states have to be performed. This causes a poor runtime performance.

To deal with this problem, we save all assignments and subgraphs of the models with constant cardinalities in hash tables. This yields an increase in runtime efficiency, where the detection rate stays equal. We also save the subgraphs with constant cardinalities, because these sub structures are often redundant in models. For instance, Figure 9(a) shows a possible model of an arbitrary object. There we assign the elements A1 and A2 with the Hungarian-Method and if the cardinality between the root node R and the node A is constant we will save the whole subgraph in a hash table.

Accordingly we exchange the subgraph with a single node, which reduces the complexity of the model. Iteratively we continue the analysis process (see FIG. 9(b))



(a)Model graph of an arbitrary object. (b)Reduced model graph.

FIG. 9: Principal of Caching of subgraphs.

until all probable models, where the belief of the model exceeds some threshold, are completely assigned or failed to assign.

Moreover, we cache the results of the methods which find the occurrences in the image and calculate the beliefs.

IV. EXPERIMENTAL EVALUATION

This section deals with the data acquisition and the experiments.

A. Image Database

The data set of domino tile images was created with a turn table in a lightbox (JUST Pantone Colour Viewing box 1). In contrast the data set of poker cards was created on a gambling table with a reflex camera. The images are made with three level of perspective distortion depending on the angle of the camera to the object (strong: around 30°, weak: around 45°, no: around 90°).

For the experiments, a data set containing 489 images

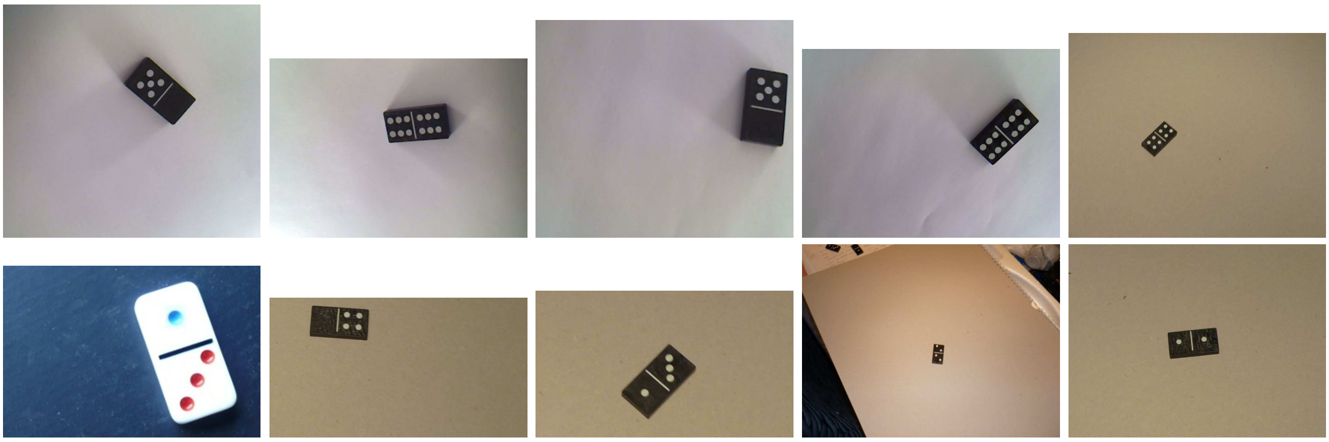


FIG. 10: Example test images of domino tiles

FIG. 11: Example test images of poker card.

of single domino tiles on a homogeneous background (see FIG. 10) and 479 images of single poker cards on a gambling table (see FIG. 11) were used (The data sets are published on: [7].).

B. Caching Results

The runtime is quite better with caching, as you can see in figure 12. Here you can see the runtime measurement of case study 1 [WHP10], this case study and only using the Hungarian-method, where the higher the model complexity is, the more model parts have to be found in the image. This shows that for models with less complexity the runtime difference is marginal, but the difference increases with increasing complexity up to a factor of three. So the caching is worthwhile.

Figure 13 shows, that method caching is the key add-on for caching. It decreases the runtime up to a factor five. Subtree caching does not make any difference in this case study. This is not surprising, because only domino tiles have subgraphs of a constants cardinality. So we have 28 models with such subgraphs, which carries no weight in comparison to thousands of states arising during the analysis.

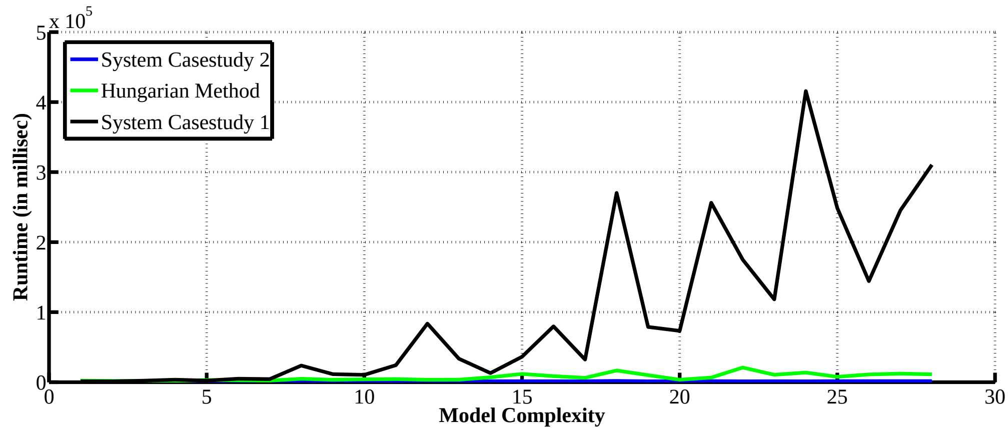


FIG. 12: Runtime measurement of case study 1, case study 2 and only using the Hungarian-method. The higher the model complexity is, the more model parts have to be found in the image. Black is case study 1 (top), green is only Hungarian-method (middle) and blue is this complete case study (bottom). We used a Intel(R) Core(TM)2 Duo 2.53GHz with 1.9 GiB working memory.

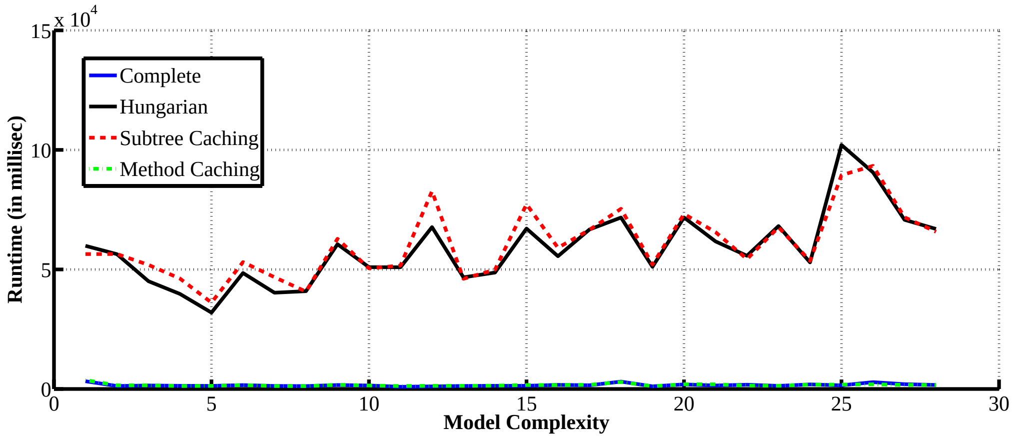


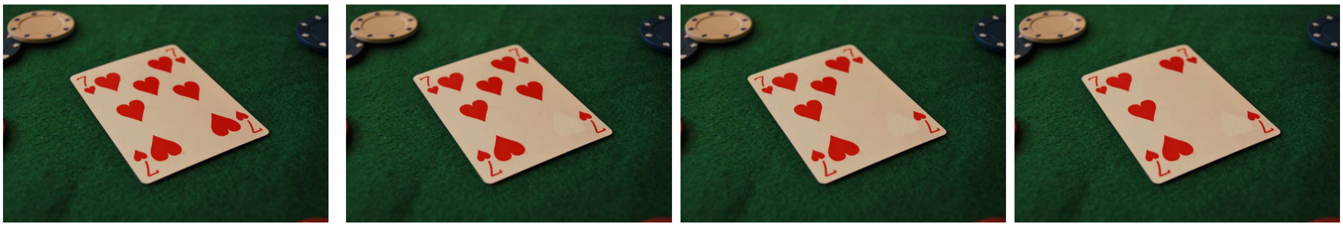
FIG. 13: Comparison of all used caching add-ons. The higher the model ID is, the more complex is the model. Black is only Hungarian-method (top), red doted is Subtree caching including the Hungarian-method(top), method caching including the Hungarian-method is green doted (bottom) and blue is using all add-ons (bottom). We used a Intel(R) Core(TM)2 Duo 2.53GHz with 1.9 GiB working memory.

C. Detection Results

Figure 14(a) shows one card of the test data set. The two most probable found classes for this card are the heart seven with a belief of 79.6% and heart six with a belief of 76.1%. If we remove than one heart (FIG. 14(b)), the two most probable classes are again heart seven (74.1%) and heart six (73.1%). Also with removing two hearts (FIG. 14(c)) the classification is correct with a belief of 68.6%. But if we remove the heart in the middle, which is characterizing for the heart seven, the classification fails. Then the most probable class is the heart four with a belief of 71.3%.

	recall	precision	detection
Poker figure cards	98.0%	94.4%	92.7%
Poker picture cards	98.0%	89.0%	87.4%
All poker cards	98.0%	93.2%	91.5%
Domino tiles	90.4%	97.8%	88.6%

TABLE I: Recall, precision and detection rate for poker cards and domino tiles. The poker cards are separated in figure and picture cards.



(a)Heart-7 of the data set.probable (b)One heart is missing.probable (c)Two hearts are missing.probable (d)Three hearts are missing.fails

FIG. 14: Example where the Poker card detection succeeds and where it does not.

The overall detection rate of domino tiles and poker cards is good with 90.6% for the whole datase. A reason for the minor performance of the classification of domino tiles in comparison to poker cards is that a domino tile has a real 3D structure and less feature than poker cards. The results (see table I) of the poker cards are (recall of 98.0% and a detection rate of 91.5%) promising that this case study is extendable to the more complex area of building recognition. This case study shows that the advantage of the approach is that it handles missing and imprecise data very well and uses a segmentation algorithm for specific regions, so that the segmentation results are better than the results of a segmentation concerning the whole image.

V. CONCLUSION

We introduced in this case study an approach which uses symbolic descriptions to recognise 2D objects in perspective images. For this purpose we combined the search in the state space with a graph-matching algorithm to deal with numerous hypotheses. All models and their assignments are rated with Dempster-Shafer beliefs, so that we deal with uncertainties and are able to tell how probable a model is correct detected. Furthermore we use caching functions for nodes and subgraphs taking advantage of the redundancy between the models to increase the runtime efficiency. We show that (i) knowledge-based object recognition with symbolic description works pretty well with perspective images and achieves in this case study a overall detection rate of 90.6%, (ii) using the Hungarian Method decrease the runtime of the system, (iii) using caching for functions is worthwhile and (iv) the approach is able to deal with uncertainties.

In a next step we will extend this approach to 3D-building recognition, where for example the facade recognition has some similarities to this case study. Facades have usually a rectangular shape and consists of windows and doors which characterise the facade, in some way comparable to a card with its card-colours and containing characters.

Acknowledgments

We thank Jürgen Ebert for the fruitful discussions on the reference schemata and the component system. Furthermore, we thank the the group of Lutz Priese for the permission and help to use their 3D campus-model. In particular, we thank the Deutsche Forschungsgemeinschaft(DFG) who partially funded this work under grant PA 599/10-1.

BHJ+99 M. Bollmann, R. Hoischen, M. Jesikiewicz, C. Justkowski, and Bärbel Mertsching. Playing domino: A case study for an active vision system. In *Computer Vision Systems*, page 392 ff., 1999.

BKI06 C. Beierle and G. Kern-Isberner. *Methoden wissensbasierter Systeme - Grundlagen, Algorithmen, Anwendungen*. Vieweg, 1. Auflage, 2006.

CL97 D. Crevier and R. Lepage. Knowledge-based image understanding systems: A survey. *Computer Vision and Image Understanding: CVIU*, 67(2):161–185, 1997.

CSS06 Daniel Cremers, Nir Sochen, and Christoph Schnörr. A multiphase dynamic labeling model for variational recognition-driven image segmentation. *International Journal of Computer Vision*, 66(1):67–81, 2006.

Ebe08 Jürgen Ebert. Metamodels taken seriously: The tgraph approach. In Kostas Kontogiannis, Christos Tjortjis, and Andreas Winter, editors, *12th European Conference on Software Maintenance and Reengineering*, Piscataway, NJ, 2008. IEEE Computer Society.

ERW08 Jürgen Ebert, Volker Riediger, and Andreas Winter. Graph technology in reverse engineering, the tgraph approach. In Rainer Gimnich, Uwe Kaiser, Jochen Quante, and Andreas Winter, editors, *10th Workshop Software Reengineering (WSR 2008)*, volume 126, pages 67–81, Bonn, 2008. GI.

Fal10 Kerstin Falkowski. A component concept for scientific experiments - focused on versatile visual component assembling. In *Proceedings of the Fifteenth International Workshop on Component-Oriented Programming (WCOP) 2010*, 2010.

FE09a Kerstin Falkowski and Jürgen Ebert. Graph-based urban object model processing. In Uwe Stilla, Franz Rottensteiner, and Nicolas Paparoditis, editors, *Object Extraction for 3D City Models, Road Databases and Traffic Monitoring - Concepts, Algorithms and Evaluation (CMRT) 2009*, volume 38 - 3 / W4, pages 115 – 120. International Society for Photogrammetry and Remote Sensing, 2009.

FE09b Kerstin Falkowski and Jürgen Ebert. The stor component system. Technical Report 14/2009, Institut für Softwaretechnik, Universität Koblenz-Landau, 2009.

HN08 Lothar Hotz and Bernd Neumann. High-level expectations for low-level image processing. *KI 2008: Advances in Artificial Intelligence*, pages 87–94, 2008.

Hoi07 Johana Hois. Towards combining ontologies and uncertain knowledge. *progic07: The Third Workshop on Combining Probability and Logic*, 2007. University of Kent, Canterbury, UK.

Hoi10 Joana Hois. Modularizing spatial ontologies for assisted living systems. *Knowledge Science, Engineering and Management*, pages 424–435, 2010.

Hu62 M.K. Hu. Visual pattern recognition by moment invariants. *IEEE Transactions on Information Theory*, 8(2):179–187, 1962.

HZ03 Richard I. Hartley and Andrew Zisserman. *Multiple*

View Geometry in Computer Vision. Cambridge University Press, 2 edition, 2003.

KSN92 F Kummert, G. Sagerer, and H. Niemann. *A problem-independent control algorithm for image understanding*, pages 297–301. Universität Bielefeld, 1992.

Kuh55 Harold W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2:83–97, 1955.

MAD09 E. Michaelsen, Michael Arens, and Leo Doktorski. Interaction of control and knowledge in a structural recognition system. In *Proceedings of the 32nd annual German conference on Advances in artificial intelligence*, pages 73–80. Springer-Verlag, 2009.

Neu08 B. Neumann. Bayesian compositional hierarchies - a probabilistic structure for scene interpretation. Technical Report FBI-HH-B-282/08, Department of Informatics, Hamburg University, 2008.

Nie90 Heinrich Niemann. *Pattern Analysis and Understanding*, volume 4. Springer Verlag, Heidelberg, 1990.

NSSK90 H. Niemann, G. Sagerer, S. Schröder, and F. Kummert. Ernest: A semantic network system for pattern understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9:883–905, 1990.

Qui95 Franz Quint. Aerial image understanding using digital map based semantic models. In *Euroconference GIS, Union Europeenne, Project Capital Humain et Mobilite, ENSG*, Saint-Mande, France, 1995.

Qui97 Franz Quint. *Kartengestützte Interpretation monokularer Luftbilder*. PhD thesis, Universität Fridericiana zu Karlsruhe (TH), Karlsruhe, 1997.

RSH09 Thomas Reineking, Niclas Schult, and Joana Hois. Evidential combination of ontological and statistical information for active scene classification. In *KEOD*, 2009.

Sag85 G. Sagerer. *Darstellung und Nutzung von Expertenwissen für ein Bildanalysesystem*. Springer, Berlin, 1985.

SDP09 J Schmittwilken, D. Dörschlag, and L. Plümer. Attribute grammar for 3d city models. In *Urban and regional data management: UDMS annual 2009: Proceedings of the Urban Data Management Society symposium 2009, Ljubljana, Slovenia, 24-26 June 2009*, page 49. CRC Press, 2009.

Sha76 Glenn Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, Princeton, 1976.

SLS06 Edgar Seemann, Bastian Leibe, and Bernt Schiele. Multi-aspect detection of articulated objects. *onference on Computer Vision and Pattern Recognition - Volume 2 (CVPR'06)*, 2:1582–1588, 2006.

TK09 Sergios Theodoridis and Konstantinos Koutroumbas. *Pattern Recognition*. Academic Press, 4 edition, 2009.

WHP10 Stefan Wirtz, Marcel Häselich, and Dietrich Paulus. Model-based recognition of domino tiles using tgraphs. In *Pattern Recognition, Proceedings of the 32nd DAGM*, pages 101–110, 2010.

1 `http://www.opengeospatial.org/standards/kml`

2 `http://www.opengeospatial.org/standards/gml`

3 `http://www.w3.org/2004/OWL/`

4 `http://www.uni-koblenz.de/~puma`

5 `opencv.willowgarage.com/`

6 `code.google.com/p/javacv/`

7 `er.uni-koblenz.de`

APPENDIX A: AUTHOR BIOGRAPHY

Stefan Wirtz obtained a diploma in Biomathematics (Dipl.-Math. (FH)) from the University of applied science RheinAhrCampus Remagen in 2008. Hi is now working for the Institute of Computational Visualistics in the Working Group Active Vision (Prof. Paulus) at the University of Koblenz-Landau since 2009. There he works as PhD student in the project “Software Techniques for Object Recognition (STOR)” which is funded by the German Research Foundation (DFG). His scientific interests can be associated with the fields of Image Processing, Pattern Recognition and especial the handling of uncertain knowledge.

Kerstin Falkowski obtained a diploma in Computational Visualistics (Dipl.-Inform.) from the University of Koblenz-Landau, Koblenz, Germany in 2005. Since then she is working at this university as PhD student for the Institute of Software Technology in the Working Group of Prof. Ebert. There she works in the project “Software Techniques for Object Recognition (STOR)” which is funded by the German Research Foundation (DFG). Her scientific interests are graph-based knowledge processing and component concepts.

Dietrich paulus obtained a Bachelor degree in Computer Science from University of Western Ontario, London, Canada, followed by a diploma (Dipl.-Inf.) in Computer Science and a PhD (Dr.-Ing.) from Friedrich-Alexander University Erlangen-Nuremberg, Germany. He obtained his habilitation in Erlangen in 2001. Since 2001 he is at the institute for computational visualistics at the University Koblenz-Landau, Germany where he became a full professor in 2002. He is head of the Active Vision Group (AGAS). His primary interests are computer vision and robot vision.