Model-Based Recognition of Domino Tiles using TGraphs

S. Wirtz, M. Häselich, D. Paulus *

University of Koblenz-Landau
Department for Computer Science

Abstract. This paper presents a case study showing that domino tile recognition using a model-based approach delivers results comparable to heuristic or statistical approaches. The knowledge on our models is modeled in TGraphs which are typed, attributed, and ordered directed graphs. Four task-independent rules are defined to create a domain independent control strategy which manages the object recognition. To perform the matching of elements found in the image and elements given by the model, a large number of hypotheses may arise. We designed several belief functions in terms of Dempster-Shafer in order to rate hypotheses emerging from the assignment of image to model elements. The developed system achieves a recall of 89.4% and a precision of 94.4%. As a result we are able to show that model based object recognition is on a competitive basis with the benefit of knowing the belief in each model. This enables the possibility to apply our techniques to more complex domains again, as it was tried and canceled 10 years ago.

1 Introduction

Object detection using model-based image analysis is still a challenge. Although there has been research in this area for many years. It is widely excepted that knowledge of the object domain is needed to classify objects. For example, Google Earth provides lots of models of buildings which are designed in Geography Markup Language (GML) and Keyhole Markup Language (KML). These model descriptions deliver symbolic descriptions of buildings which may give new relevance to model-based image analysis, because plenty of models are available.

Several approaches try to find an algorithm which is general, robust and efficiently calculable. In this study we evaluate the *efficiency* and *robustness* of model-based object recognition in a *task-independent* pattern recognition system. Furthermore we introduce *belief functions* to achieve that goal. For this purpose we use exemplary domino tiles as a first manageable application domain.

One archetype of such a system is ERNEST [1], which is a pattern recognition system developed in 1980 employing semantic networks in several application domains, like building recognition [2] and speech understanding [3]. In

* This work was partially funded by Deutsche Forschungsgemeinschaft (DFG) under grant PA 599/8-1.

our approach, we use TGraphs [4] as a "light-weight"-ontology. TGraphs are a very general graph concept which is based on vertices and edges as first-class entities and includes types, attributes, and ordering for both. We store the meta-knowledge and instantiate concrete models using this knowledge, taking advantage of the fact that no large sets of samples are needed because the geometrical knowledge about domino tiles is directly used to construct the models. To find features in images we exploit the explicit knowledge given by the models. Therefore, we use a task-independent *activity-control*, which manages the application flow of the system. In this paper, we outline how our model-based system can be extended to work in any other application domain.

We introduce the related work in Section 2. Section 3 describes our approach in detail. There we give the exact definition of the task and describing our model of domino tiles as well as the designed belief functions. We show and discuss our experiments and results in Section 4. A summary and ideas for future work can be found in Section 5.

2 Related Work

In principle, the object detection task for domino tiles is simple; this problem has been used as a case study before, e.g. in [5]. A heuristic strategy is to detect circles and rectangles and count the circles in each half of the domino tile. Another approach is to apply *template matching* [6], which is widely used in face recognition. Bollmann et. al. [5] use template matching to identify domino tiles. The advantage of template matching is that it delivers a similarity measure, but it is neither rotation nor scale invariant.

Today, model-based approaches normally use statistics, appearance, shape or geometry, which mostly work without an explicit geometrical model schema [7, 8]. In this work we only use models where the modeling of the objects is in a explicit form, like it is used in Computer Aided Design, which rather corresponds to the human view. We use domino tiles as an example to demonstrate and evaluate model-based approaches.

Using models for explicit knowledge representation, there are two main strategies how a controlling algorithm can handle the analysis process (cf. [9] p. 240ff). On the one hand, there is the *data-driven* strategy, where the segmentation objects, found in the image, serve as an initialization for the analysis. Based on the segmentation objects, the best possible model is sought. The *model-driven* strategy on the other hand works the other way round. Each model determines whether it is contained in the image or not and tries to locate its elements.

Hybrid forms are feasible and favored in most cases. We choose a hybrid approach where data-driven models become pre-evaluated and then we selectively search model-driven for segmentation objects. These strategies can be called task-independent because they do not refer to knowledge of the domain. Other systems exist which use ontologies such as OWL[1] and combine them with uncertain knowledge for finding concepts in a domain [10, 11].

[1] http://www.w3.org/2004/OWL/

Uncertainty theories, such as Bayes [6], Dempster-Shafer [12] or fuzzy sets [12] 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 [13] and [10], where Neumann uses Bayesian compositional hierarchies for scene interpretation [14].

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 [15–17]. The formalism ERNEST [17] 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. This can be controlled by the A^*-algorithm [15].

An overview of knowledge-based systems for object recognition is given in [18]. Such systems need to be robust to errors in segmentation data as well as predominantly invariant to changes in image acquisition.

3 The Model-Based Approach

This paper describes a model-based approach to locate domino tiles in images, where Section 3.1 gives a formal specification of the task. The application flow of the system is to accomplish first a pre-evaluation of the models (3.2), where only the circles in each half of a rectangle are counted. Then the activity control (3.3) starts to recognize domino tiles with these model. The hypotheses of found domino tiles are evaluated for each assignment (3.4) and the entire model.

3.1 Specification

The recognition system in this case study works with several domino tiles and can handle wrong inputs.

Given

- a universe SEGMENT of all possible segmentation objects,
- a set RECT $\subset$ SEGMENT of all possible rectangles,
- a set CIRCLE $\subset$ SEGMENT of all possible circles,
- $\mathrm{CIRCLE} \cap \mathrm{RECT} = \varnothing$
- a relation CONTAINS which describes if a segmentation objects contains another segmentation objects, $\mathrm{CONTAINS} \subseteq \mathrm{SEGMENT}\times\mathrm{SEGMENT}$,
- one domino tile rectangle $\mathrm{dr}_1 \in \mathrm{RECT}$ and its two halves $\mathrm{dr}_2 \in \mathrm{RECT}$ and $\mathrm{dr}_3 \in \mathrm{RECT}$,
- a set $\mathrm{P} \subset \mathrm{CIRCLE}$ of possible pips with a specific position, with $|\mathrm{P}| = 14$,
- a set LAYOUT of possible layouts, $|\mathrm{LAYOUT}| = 28$, the number of possible domino tiles, and $\mathrm{LAYOUT} \subset 2^{\mathrm{P}}$,
- a set of Models $\mathrm{M} = \{\mathrm{dr}_1\}\times\{\mathrm{dr}_2\}\times\{\mathrm{dr}_3\}\times\mathrm{LAYOUT}$, where $m \in \mathrm{M} \wedge m = (m_1, m_2, m_3, m_4)$ is a concrete model (see Sect. 3.2).

Input

- a set $\mathrm{S} \subset \mathrm{SEGMENT}$, $\mathrm{S} = \mathrm{R} \dot{\cup} \mathrm{C}$ with $\mathrm{R} \subset \mathrm{RECT}$ and $\mathrm{C} \subset \mathrm{CIRCLE}$, of all segmentation objects found in the image,
- a threshold $\theta \in [0, 1]$, to set a belief limit of the hypotheses
- a total function $\mathrm{Bel_S} : \mathrm{S} \to [0, 1] \subset \mathbb{R}$, describing the belief in the segmentation object itself
- a total function $\mathrm{Bel_M} : \mathrm{M} \times (2^{\mathrm{R}} \times 2^{\mathrm{C}}) \to [0, 1] \subset \mathbb{R}$, describing the belief in the model assignment.

Output

- a set $\mathrm{M}' \subset \mathrm{M}$ of domino models, where $\forall\, m' \in \mathrm{M}' \mid \mathrm{Bel_M}(m', \mathrm{R}', \mathrm{C}') \geq \theta$, with $\mathrm{C}' \in 2^{\mathrm{C}} \wedge \mathrm{R}' \in 2^{\mathrm{R}}$,
- $\forall\, m' \in \mathrm{M}'$
  - a set $\mathrm{R}^{m'} \subset \mathrm{R}$ of rectangles partially mapped to m'_1, m'_2 and m'_3, with $|\mathrm{R}| < 4$, where $r_1 \in \mathrm{R}^{m'}$ is assigned to the domino tile m'_1,
  - $\gamma_{\mathrm{C}}^{m'} = \mathrm{C} \rightarrowtail\!\!\!\!\!\!\!+\;\, m'_4$,
  - $\forall\, c_1 \in \mathrm{C} \mid c_1 \in \mathrm{dom}(\gamma_{\mathrm{C}}^{m'}) :\ r_1\ \mathrm{CONTAINS}\ c_1$
  - judgments $\mathrm{J}_{\mathrm{local}_k}^{m'} \in [0, 1],\ k = 1, \ldots, \mathrm{L}$ how well these mappings in detail could be established,
  - a total judgment $\mathrm{J}^{m'} \in [0, 1]$ of the found models. These are calculated with $\mathrm{Bel_M}$.

In this case study the judgment J is according to the belief of the Dempster-Shafer theory [19].

3.2 Model

The basis for the model-based object recognition system is, of course, the model itself. Therefore, we model the declarative knowledge in a generic scheme (see Fig. 1). The main elements of the model are the elements `TwoDGeometricObject` and `SemanticObject`, which are derived classes of the `GenericObject`, so we have a clear separation of semantic and geometric objects.

Two types of edges are defined in the schema. On the one hand, the `consists-Of` edge which describes that `TwoDGeometricObjects` are able to consist of other `TwoDGeometricObjects` and `SemanticObject` are able to consist of other `SemanticObjects`. A constraint to the schema is that `SemanticObjects` cannot consist of `TwoDGeometricObjects` and the other way round. On the other hand, the `isRepresentedBy` edge maps a `SemanticObject` to a `TwoDGeometricObject`. Every edge has the attribute `obligatory` to determine if the part or representation is obligatory to recognize the object.

In addition to declarative knowledge, we also model the procedural knowledge in the schema. Therefore, we need the three methods `initiate`, `limit` and `calculateBelief` for every element. The `initiate` method has the task to find its potential equivalents in the image and to provide them to the system. After

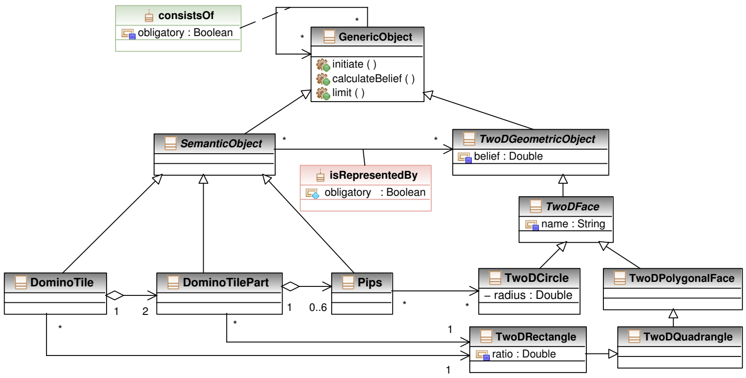


Fig. 1. UML diagram of the domino tile schema.

a successful initialization, the `limit` method is needed to calculate with a given segmentation object, as initiation candidate, the resulting limiting, region of interest (*ROI*). This ROI is set to the further parts of the model, so that in further calculation steps only candidates within the ROI are considered. Finally, the `calculateBelief` method calculates the belief in the element. These belief functions vary widely and it is a scientific task to find adequate belief functions. In Section 3.4, the functions are described in more detail.

With this schema, we are able to construct concrete models. In this case study, we have 28 basic models, because we have 28 domino tiles.

3.3 Pragmatic of the System

A control system can now take these models and start with the analysis of the image. The developed system is inspired by ERNEST (see Sect. 2). In ERNEST six task-independent rules precisely define the computation of elements and instances. These rules are combined with graph search algorithms to handle the control problem. Our system has four task-independent rules.

Rule 1 If a `isRepresentedBy` edge belongs to the actual element, go to the `TwoDGeometricObject` of the `isRepresentedBy`-edge and initiate the element.

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 initialization of a element was successful, limit the ROI of all parts which belong to this element.

Rule 4 If an element is not obligatory, but an initialization is possible, then use Rule 1 for this element.

In our model only the `TwoDRectangles` of the `DominoTileParts` are not obligatory, because we are able to identify a domino tile without these rectangles, but the knowledge of them gives a additional contribution in the belief that we have found a domino tile in the image.

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

3.4 Belief Functions

With the models and the activity control we are able to instantiate them, 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* (bpa). A bpa must fulfill the conditions:

Given the sample space $\Omega = \{a_k\}, k = 1, \ldots, K, A \subseteq \Omega$, where a_k and A are arbitrary events.

$$\mathrm{bpa}(\varnothing) = 0 \qquad (1) \qquad\qquad \sum_{A \subseteq \Omega} \mathrm{bpa}(A) = 1 \qquad (2)$$

The combined bpa with Dempster-Shafer is:

$$\mathrm{bpa}_1 \oplus \mathrm{bpa}_2 = \mathrm{bpa}(A) = \begin{cases} 0 & : \quad A = \varnothing \\ \frac{\sum_{t \cap u = A} \mathrm{bpa}_1(t)\mathrm{bpa}_2(u)}{1 - \sum_{t \cap u \neq \varnothing} \mathrm{bpa}_1(t)\mathrm{bpa}_2(u)} & : \quad \varnothing \neq A \subseteq \Omega \end{cases} . \tag{3}$$

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 bpa $\oplus$ bpa $\neq$ bpa is valid.

We define three general functions which are valid for each `SemanticObject`. At first we need a function to combine bpa of two events of different sample spaces which gives both a belief assignment that an accordant element is found in the image (see eqn. 4). Therefore, the combination rule of DS (eqn. 3) is used like Quint made it in [2]. Then the belief of a `SemanticObject` is

$$\begin{aligned} &\mathrm{bpa}(\texttt{SemanticObject} \mid \mathrm{E}_{parts}, \mathrm{E}_{rep}) = \\ &\quad \kappa_1 \ \mathrm{bpa}(\texttt{SemanticObject} \mid \mathrm{E}_{parts}) \oplus \kappa_2 \ \mathrm{bpa}(\texttt{SemanticObject} \mid \mathrm{E}_{rep}) \ . \end{aligned} \tag{4}$$

With κ_1 and κ_2 you can weight the trust that this event really support 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 which represents this `SemanticObject`. A example is given in equation 5 with the belief

$$\mathrm{bpa}(\mathrm{dr}_1 \mid \mathrm{E}_{\mathrm{dr}_i}, \mathrm{E}_\mathrm{R}) = \quad 0.7\mathrm{bpa}(\mathrm{dr}_1 \mid \mathrm{E}_\mathrm{R}) \oplus \mathrm{bpa}(\mathrm{dr}_1 \mid \mathrm{E}_{\mathrm{dr}_i}), i \in \{2, 3\} \tag{5}$$

of the `SemanticObject DominoTile` given the information sources $\mathrm{E}_{\mathrm{dr}_i}$ (observed domino tile parts) and E_{R} (observed rectangle). The rectangle alone gives a smaller contribution that the `DominoTile` element was found in the image than possibly found domino tile halves, so $\kappa_1 = 0.7 \wedge \kappa_2 = 1$. Also we need to know the belief of a `SemanticObject` given the information source E_{parts} (observed parts). Therefore, we use a function (eqn. 6) equal to the total-probability rule. For any partition $\mathrm{B}_j,\ j = 1, \ldots, \mathrm{N}$ of the event space Ω and $\Omega = \{\{\mathrm{fp}_1, \ldots, \mathrm{fp}_\mathrm{L}\}, \{\overline{\mathrm{fp}_1}, \ldots, \mathrm{fp}_\mathrm{L}\}, \ldots, \{\overline{\mathrm{fp}_1}, \ldots, \overline{\mathrm{fp}_\mathrm{L}}\}\}$, where fp equates the event of found a `SemanticObject` part, is

$$\begin{aligned}&\mathrm{bpa}(\texttt{SemanticObject} \mid \mathrm{E}_{\mathrm{parts}}) = \\ &\quad\sum_j \mathrm{bpa}(\texttt{SemanticObject} \mid \mathrm{B}_j)\mathrm{bpa}(\mathrm{B}_j \mid \mathrm{E}_{\mathrm{parts}})\ .\end{aligned} \tag{6}$$

We apply the belief of a `SemanticObject` given the information source E_{parts} to a `DominoTile` as example (eqn. 7) and calculate the weighted sum over the partition B of the `DominoTileParts`. For this is $\Omega = \{\{\mathrm{dr}_2, \mathrm{dr}_3\}, \{\overline{\mathrm{dr}_2}, \mathrm{dr}_3\}, \{\mathrm{dr}_2, \overline{\mathrm{dr}_3}\}, \{\overline{\mathrm{dr}_2}, \overline{\mathrm{dr}_3}\}\}$ and

$$\mathrm{bpa}(\mathrm{dr}_1 \mid \mathrm{E}_{\{\mathrm{dr}_2,\mathrm{dr}_3\}}) = \sum_j \mathrm{bpa}(\mathrm{dr}_1 \mid \mathrm{B}_j)\mathrm{bpa}(\mathrm{B}_j \mid \mathrm{E}_{\{\mathrm{dr}_2,\mathrm{dr}_3\}})\ . \tag{7}$$

Finally, we need the belief of a partition B_j given the information source $\mathrm{E}_{\mathrm{B}_j}$:

$$\mathrm{bpa}(\mathrm{B}_j \mid \mathrm{E}_{\mathrm{B}_j}) = \prod_{p_j \in \mathrm{B}_j \wedge p_j \,\mathrm{isTrue}} \mathrm{bpa}(p_j)\ . \tag{8}$$

The concrete implementation for the partition B_j of domino tile parts is

$$\mathrm{bpa}(\mathrm{B}_j \mid \mathrm{E}_{\{\mathrm{dr}_2,\mathrm{dr}_2\}}) = \prod_{\mathrm{dtp}_j \in \mathrm{B}_j \wedge \mathrm{dtp}_j \,\mathrm{isTrue}} \mathrm{bpa}(\mathrm{dtp}_j)\ . \tag{9}$$

An example of a possible partion B_j is $\mathrm{B}_j = \{\overline{\mathrm{dr}_2}, \mathrm{dr}_3\}$.

These three functions are the same for each `SemanticObject`. They have to be supplemented by basic probabilistic assignments bpa for the specific cases. Some important bpa will be presented in the next equations 10 - 13.
The belief of the assignment of a segmented rectangle to a model rectangle is

$$\mathrm{bpa}(\mathrm{dr}_1 \mid \mathrm{E}_\mathrm{R}) = \begin{cases} \frac{\mathrm{ratio}_r}{\mathrm{ratio}_{\mathrm{dr}_1}} & : \ \mathrm{ratio}_{\mathrm{dr}_1} > \mathrm{ratio}_r, \quad r \in \mathrm{R} \\ \frac{\mathrm{ratio}_{\mathrm{dr}_1}}{\mathrm{ratio}_r} & : \ \mathrm{ratio}_{\mathrm{dr}_1} < \mathrm{ratio}_r, \quad r \in \mathrm{R} \end{cases}\ . \tag{10}$$

For the calculation of $\mathrm{bpa}(\mathrm{dr}_i \mid \mathrm{E}_\mathrm{R})$ we take the already found rectangle dr_1 of the domino tile, divide it in two halves and compare it with the candidates for the domino tile part rectangle dr_i. Therefore, we calculate the intersection area of one of the halves and dr_i and divide the intersection by the entire area of the rectangle half and dr_i.

$$\mathrm{bpa}(\mathrm{dr}_i \mid \mathrm{E}_\mathrm{R}) = \frac{\text{intersection area of } \mathrm{dr}_i \text{ and R}}{\text{entire area of } \mathrm{dr}_i \text{ and R}} \text{ with } i \in \{2, 3\}\ . \tag{11}$$

The beliefs of the rectangle assignments in (10) and (11) are necessary to distinguish a domino tile from other objects. We also need a belief function which penalize missing assignments of segmentation objects and/or model elements:

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

The belief in the assignment of a pip p to a circle c, where (c_x, c_y) and (p_x, p_y) are the center points of the circles, and c_r and p_r are the radiuses, is very important. We choose the exponential form to prevent that circles are assigned to wrong and far away pips:

$$\begin{aligned} \mathrm{bpa}(p \in \mathrm{P} \mid c \in \mathrm{C}) &= 101 - 200^q - 50 \mid c_r - p_r \mid, \\ q &= \sqrt{(p_x - c_x)^2 + (p_y - c_y)^2}\ . \end{aligned} \tag{13}$$

The choice of the functions (12) and (13) tunes if it is better to drop an assignment or not. As much more missing elements are penalized and as much more the bpa($p \in \mathrm{P} \mid c \in \mathrm{C}$) forgive differences of pips and circles, the more segmentation objects are assigned and the other way round. For the calculation of the beliefs (11), (12) and (13) we translate and rotate the rectangle and circles in the origin and scale them, so they are comparable with model elements.

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

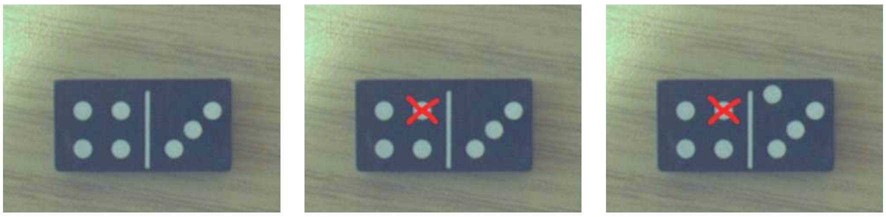
4 Experiments and Results

This section deals with the data acquisition and the conducted experiments.

The data set of images was created with a turn table in a lightbox (JUST Pantone Color Viewing box 1). For the experiments, a data set containing 489 images of single domino tiles on a homogeneous background was used. The images were created with an standard USBcamera in an approximate orthogonal angle to the objects. Furthermore, an adjustment of 22-45 degrees per rotation step (8, 12 or 16 images per light source) was selected for the turn table. In addition to the automatically generated images we also added images with sunlight and ambient light to the data set[2].

We achieved a detection rate of 94.7% and a precision rate of 94.4%. This proves well, that our system works accurately with the given models, and found rectangles and circles. A recall of 89.4% also seems acceptable and shows that the classification is in balance of precision and recall. The Fig. 2(a) shows one image of the database where our classification correctly recognize the domino tile

[2] The dataset is published on er.uni-koblenz.de.



(a) (b) (c)

Fig. 2. Image (a) is a possible image with sunlight of the database. This image is additionally distorted to test the classification. Therefore, a circle is masked (b) (c) and a circle is added (c).

as a 4-3 domino tile with a belief of 97.3%. We manually distorted this image, where in Fig. 2(b) one circle is not detectable and in Fig. 2(c) a additional circle is added. For Fig. 2(b) the classification delivers as result a 4-3 domino tile with a belief of 84.9%. Hence the classification recognizes the correct domino tile, even though the segmentation is incomplete because we weight the position of a circle more than the number of circles. Whereas for Fig. 2(c) the classification failed. There the classification delivers as result a 4-4 domino tile with a belief of 68,9% because it is more likely that the circle is only at the wrong position than it is a false segmentation, especially where the radius of this circle is correct. As expected, the belief in the result is low. Furthermore, we know the belief of the single assignments. So the belief in the assignment of the wrong circle to the pip is 41.5%, where the belief of a correct circle to a pip is about 98%.

5 Conclusion

We demonstrated that a model-based approach with a task-independent control is able to deliver solid and accurate results for the recognition of domino tiles in images. We achieved a recall of 89.4% and a precision of 94.4%. We showed that we can not only calculate all assignments, but also obtain the knowledge which elements of an image is the best correspondence for an element among the model. To adapt this strategy to another application domain we have to create a new model and implemented the methods for initialization, belief calculation and limitation. Afterwards the activity control is able to classify. In the moment we aply this approach in order to classify poker cards and in the near future the evolved strategies will be adapted to more complex problems, such as the classification of traffic signs and buildings, which will continue the work that we reported on in [20].[3]

[3] We thank Kerstin Falkowski and Jrgen Ebert for their fruitful discussions on the component system and the specification in Sect. 3.1.

References

1. Niemann, H., et al.: ERNEST: A Semantic Network System for Pattern Understanding. **12** (1990) 883–905
2. Quint, F.: Kartengesttzte Interpretation monokularer Luftbilder. PhD thesis, Universität Fridericiana zu Karlsruhe (TH) (1997)
3. Niemann, H., et al.: Ernest: A semantic network system for pattern understanding. IEEE Trans. Pattern Anal. Mach. Intell. **12**(9) (1990) 883–905
4. Ebert, J., et al.: Graph technology in reverse engineering, the tgraph approach. In Gimnich, R., Kaiser, U., Quante, J., Winter, A., eds.: 10th Workshop Software Reengineering (WSR 2008). Volume 126 of GI Lecture Notes in Informatics., Bonn, GI (2008) 67–81
5. Bollmann, M., et al.: Playing Domino: A Case Study for an Active Vision System. In: Computer Vision Systems. (1999) 392 ff.
6. Theodoridis, S., Koutroumbas, K.: Pattern Recognition. 4 edn. Academic Press (2009)
7. Cremers, D., Sochen, N., Schnrr, C.: A multiphase dynamic labeling model for variational recognition-driven image segmentation. International Journal of Computer Vision **66**(1) (1 2006) 67–81
8. Seemann, E., Leibe, B., Schiele, B.: Multi-aspect detection of articulated objects. onference on Computer Vision and Pattern Recognition - Volume 2 (CVPR'06) **2** (2006) 1582–1588
9. Niemann, H.: Pattern Analysis and Understanding. Volume 4 of Springer Series in Information Sciences. Springer Verlag, Heidelberg (1990)
10. Hois, J.: Towards Combining Ontologies and Uncertain Knowledge. progic07: The Third Workshop on Combining Probability and Logic (2007) University of Kent, Canterbury, UK.
11. Reineking, T., Schult, N., Hois, J.: Evidential Combination of Ontological and Statistical Information for Active Scene Classification. In: KEOD. (2009)
12. Beierle, C., Kern-Isberner, G.: Methoden wissensbasierter Systeme - Grundlagen, Algorithmen, Anwendungen. Vieweg, 1. Auflage (2006)
13. Joana Hois, Kerstin Schill, J.A.B.: Integrating Uncertain Knowledge in a Domain Ontology for Room Concept Classifications. In: Research and Development in Intelligent Systems XXIII. (2007)
14. Neumann, B.: Bayesian compositional hierarchies - a probabilistic structure for scene interpretation. Technical Report FBI-HH-B-282/08, Department of Informatics, Hamburg University (2008)
15. Kummert, F., Sagerer, G., Niemann, H. In: A Problem-independent Control Algorithm for Image Understanding. Universität Bielefeld (1992)
16. Sagerer, G.: Darstellung und Nutzung von Expertenwissen fr ein Bildanalysesystem. Springer, Berlin (1985)
17. Niemann, H., et al.: Ernest: A semantic network system for pattern understanding. IEEE Transactions on Pattern Analysis and Machine Intelligence, http://www.computer.org/tpami/ **9** (1990) 883–905
18. Crevier, D., Lepage, R.: Knowledge-Based Image Understanding Systems: A Survey. Computer Vision and Image Understanding: CVIU **67** (1997) 161–185
19. Shafer, G.: A Mathematical Theory of Evidence. Princeton University Press, Princeton (1976)
20. Falkowski, K., et al.: Semi-automatic generation of full citygml models from images. In: Geoinformatik 2009. Volume 35 of ifgiPrints., Institut für Geoinformatik Westfälische Wilhelms-Universität Münster (4 2009) 101–110