

# Eye Tracking Meets the Process of Process Modeling: A Visual Analytic Approach

Andrea Burattin<sup>1(✉)</sup>, Michael Kaiser<sup>1</sup>, Manuel Neurauter<sup>1</sup>,  
and Barbara Weber<sup>1,2</sup>

<sup>1</sup> University of Innsbruck, Innsbruck, Austria  
[andrea.burattin@uibk.ac.at](mailto:andrea.burattin@uibk.ac.at)

<sup>2</sup> Technical University of Denmark, Kongens Lyngby, Denmark

**Abstract.** Research on the process of process modeling (PPM) studies how process models are created. It typically uses the logs of the interactions with the modeling tool to assess the modeler’s behavior. In this paper we suggest to introduce an additional stream of data (i.e., eye tracking) to improve the analysis of the PPM. We show that, by exploiting this additional source of information, we can refine the detection of comprehension phases (introducing activities such as “semantic validation” or “problem understanding”) as well as provide more exploratory visualizations (e.g., combined modeling phase diagram, heat maps, fixations distributions) both static and dynamic (i.e., movies with the evolution of the model and eye tracking data on top).

**Keywords:** Process of process modeling · Eye tracking · Modeling phase diagram

## 1 Introduction

The adoption of business process models is gaining significant relevance due to their importance in managing business processes [1]. Therefore, it is relevant to analyze the factors leading towards high quality models [2]. Significant research has been done to better understand the factors influencing the quality of process models [2–4]. These works, however, typically focus solely on the final outcome of the act of modeling, without considering the process of creating the process model. Instead, a more recent research field is investigating the *process of process modeling* (PPM): in this case the emphasis is on the process underlying the creation of process models [5–10]. The creation of a process model, starting from a domain description, subsumes the execution of several activities. In particular, the modeler first has to construct a mental model of the process and the expected behaviors, then she has to externalize and map the mental model into a proper representation, interacting with a modeling tool [8, 9]. In general, modeling is an iterative and highly flexible process comprising three phases that are repeatedly executed in various orders, i.e., *comprehension*, *modeling*, and *reconciliation* [5]. Different techniques have been suggested to both analyze and

visualize the PPM [7, 11, 12]. While these techniques have been demonstrated to be useful as tool for data exploration and hypothesis generation [5], their focus is on model interactions only. This might limit the analyses, since the interactions are not enough to fully reflect the activities performed by the modeler. In particular, respective techniques fall short analyzing comprehension phases, i.e., phases with no interactions with the modeling tool.

Eye tracking technologies, in turn, provide a complementary view on human behavior and allow capturing eye movements, e.g., *fixations* and *saccades*. This provides information on how visual attention is distributed and evolves within a given context. The visualization of eye tracking data serves as qualitative way of analyzing the gathered data, with the aim of finding hypotheses [13]. Besides scan path visualizations (showing sequences of fixations) and attention maps (showing fixations and their durations) a variety of visualizations for eye tracking data is used in various fields of research. However, existing eye tracking technologies are primarily tailored towards reading tasks where the content of the screen is static or only changes in response to a stimulus. The creation of process models, in contrast, constitutes a dynamic setting with constantly evolving screen content depending on user interactions. Therefore, an inter-subject comparison and hypotheses testing is not straightforward and possibilities for synchronizing and quantifying the data have to be found.

In this paper we propose to integrate both data sources (i.e., model interactions reflecting the evolution of the artifact and eye movement data highlighting the areas of attention) with the goal to provide more meaningful visualizations of the respective data. The visualizations described in this paper are suitable for depicting interactive eye tracking data gathered during process model creation as well as other human computer interaction scenarios. Moreover, the provided visualizations are potentially useful for improving existing approaches for phase detection [11] including the refinement of comprehension phases into problem understanding, method finding, as well as syntactic and semantic validation [14]. Such integrated view additionally has the potential to replay both model interactions and eye movements in an integrated manner and to build analysis techniques that allow the automatic calculation of different metrics considering the different areas of interest and the artifact being modeled (e.g., average fixation durations for modeling phases) [15]. Additional benefits of the visualizations described include validity checks on the collected data, such as biases.

Section 2 of the paper describes background information; Sect. 3 reports the new data streams and corresponding analyses; Sect. 4 describes few demonstrations; and Sect. 5 provides conclusions and implications for future research.

## 2 Background

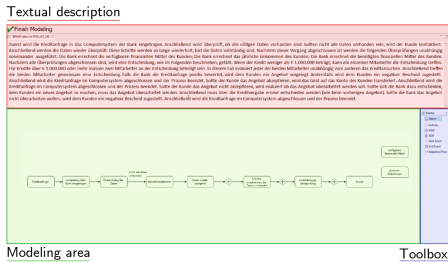
### 2.1 Visualizing the Process of Process Modeling

The process of creating a process model is a highly flexible and iterative process that involves three phases:

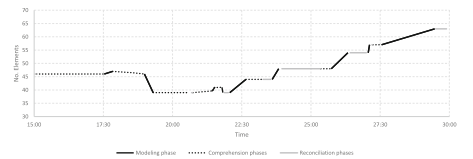
- *Comprehension*: when creating a process model, modelers first need to understand the problem (i.e., the requirements and the process model created so far). During these phases modelers build an internal representation (i.e., a mental model) of the process parts to be modeled within their working memory [9] independently of any modeling notation, which then needs to be mapped to the constructs provided by the modeling notation.
- *Modeling*: during modeling phases the modeler interacts with the modeling environment to externalize the internal representation of the problem stored in working memory. For instance, modelers might insert nodes like activities and gateways into the model and connect those using edges.
- *Reconciliation*: after a modeling phase is over, modelers might work on improving the understandability of the process model through changes to the labeling of modeling elements [3], but also layouting [16].

To enable the systematic investigation of the modeling process the research tool Cheetah Experimental Platform (CEP) has been developed [17]<sup>1</sup>. It includes a simple graphical process editor that logs all interactions of the user with the process editor (e.g., creation, deletion, renaming and moving of nodes, edges, conditions, bend points, but also scrolling) together with a timestamp. The graphical user interface of CEP is shown in Fig. 1. The textual description contains the natural language representation of the process to be modeled. The modeling canvas is the area where the modeler places all the modeling elements (i.e., activities, events and edges). The toolbox contains the items the modeler can use to translate the textual description into the graphical representation.

To automatically detect PPM phases from the logged interactions existing algorithms map the model interactions recorded by CEP onto PPM phases (i.e., comprehension, modeling, and reconciliation) [5, 11]. In general, create and delete interactions are classified as modeling phase; renaming and moving interactions characterize reconciliation phases; and comprehension phases are phases with no model interactions. The detected phases can then be plotted on the so called Modeling Phase Diagram (MPD). An example of this diagram is reported in Fig. 2. In particular,  $x$  axis reports the time, and  $y$  axis indicates the number of items currently shown on the modeling canvas.



**Fig. 1.** Cheetah modeling platform with different areas of interest.



**Fig. 2.** Modeling phase diagram. In this case only 15 min of modeling are reported.

<sup>1</sup> Cheetah Experimental Platform is available at <http://www.cheetahplatform.org>.

## 2.2 Eye Tracking

Using eye tracking techniques, researchers can gain knowledge about the information processed by subjects at any moment in time [18]. Eye tracking manufacturer’s software gathers the collected raw data to *fixations* and *saccades* in a first abstraction step. Fixations are time frames in which the gaze stops on a stimulus, lasting for about 200 ms up to several seconds. Therefore, fixations are representing the amount of time a specific area caught the subject’s attention [18]. Saccades are fast eye movements (30 to 80 ms), with the purpose of shifting the attention to different sources of information. During saccades, no information encoding takes place [19]. To analyze specific areas of the screen, areas of interest (AOI) can be defined as an additional abstraction step. Thus, deeper insights into the attention focus of the participants is possible, such as saccades from one AOI towards another AOI (called transitions) and different AOIs can be compared regarding fixation durations and fixations counts.

This type of instruments has been fruitfully used, e.g., to analyze websites’ usability [20]. By conducting eye tracking while process modeling, the actual attention focus is recorded allowing to gain knowledge about cognitive processes in place. For analyzing the data in this paper we split the screen into three AOIs (cf. Fig. 1): the textual description, the modeling canvas and the toolbox.

## 3 Analysis with Additional Data Streams

In this paper we propose to integrate the model interactions and eye movement data highlighting the areas of attention, in order to provide more meaningful visualizations of the respective data.

The MPD depicts the different high-level phases a modeler performed. However, as described, these phases are generated solely based on the list of interactions with the modeling tool [11]. While modeling and reconciliation phases can be easily derived from interaction logs, comprehension phases remain black boxes. To tackle this problem we add eye movement data as an additional event stream. This way, it becomes possible to dig into time fractions with no tool interactions and gain knowledge regarding user’s actions. For example, if during a period with no interaction, the user is just looking at the text, this might suggest that she is reading; if the modeler, in turn, focuses her attention at the model already represented, we might conclude she is validating the syntactic correctness; if the modeler’s look is jumping between the text and the model we might infer that she is semantically validating the correctness of the model with respect to the textual description of the process [14].

Note that combining the two data streams requires the time of the events to be synchronized. In our scenario, different machines were used to perform the recording of the model interactions and the recording of the eye movement data. We used the Microsoft utility `net time` to achieve synchronisation.

### 3.1 Data Visualizations

This section describes the visualizations combining model interactions and eye movement data.

**Fixations Distribution.** The first representation we would like to focus on shows the distribution of fixations on the screen (cf. Fig. 3). In this visualization, for each fixation, one black point is plotted onto the screen with the modeling environment. This visualization is useful to obtain an initial overview of the distribution of fixations. For example, Fig. 3 highlights that fixations almost entirely cover the textual description, indicating that the user focused on the entire AOI. Fixations are also observed on the toolbox and on the central area of the modeling canvas. Some fixations on the canvas scrollbar can be detected as well.



**Fig. 3.** Fixations distribution over CEP.

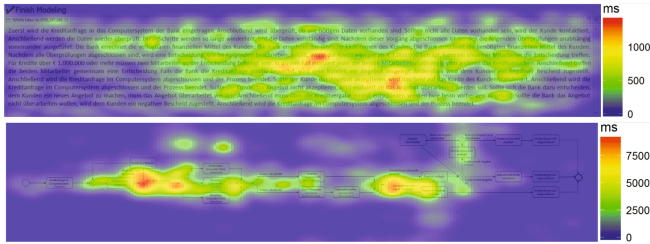
Combining fixations with the stream of model interactions enables a “smart filtering” that can be applied to the visualization of fixations. For example, it is possible to filter only the fixations occurring during comprehension phases. This visualization can also be useful to identify (systematic) biases of the eye tracker which might be a problem for an automatic phase detection (see also Sect. 4).

**Fixations Heat Map.** While the first visualization focuses on the distribution of fixations, the second visualization focuses on the time spent looking at specific parts of the screen (i.e., fixations durations) using heat maps [21]. These visualizations map values to a color scheme. In our case, the color is associated to the sum of fixations durations on each point<sup>2</sup> of an area of interest.

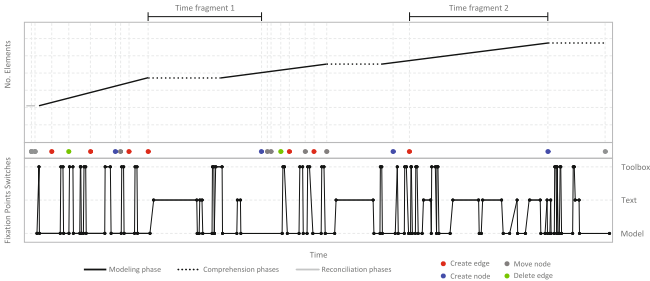
The collection of model interactions and their synchronization with the fixations enable us to compute the heat map for each AOI independently and not just for the modeling AOI that is visible to the user, but for the whole modeling canvas. The idea is that the modeling AOI can be seen as a *viewport*, showing a portion of the whole model (scrollbars are employed to host large models). Therefore, by considering the “offset” given by the scrollbars positions, we map each fixation to the actual model (which might be larger than the modeling AOI). Fixations are thereby plotted onto the final process model, i.e., the model version at the end of the modeling session. Please note that if during the creation of the model objects are moved around or deleted, the final process model is not adequate. The solution to this problem is described in Sect. 3.2.

The picture on top of Fig. 4 reports the heat map for the textual description, the one below contains the heat map of the the entire process model.

<sup>2</sup> Actually, an approximation is applied by dividing the AOIs with a “grid system”.



**Fig. 4.** Heatmap representations of the textual description and the process model.



**Fig. 5.** Combined MPD with logged events and transitions between AOIs.

**Combined Modeling Phase Diagram.** Integration of data sources allows to enhance modeling phase diagrams with additional information regarding fixations. Specifically, the idea is to introduce the list of model interaction as well as the list of fixations on each AOI using the same diagram. These different elements share the time dimension and can thus be synchronized via their timestamps. An example of such chart is depicted in Fig. 5 which reports a fragment of a complete modeling session. Thereby, the  $x$  axis refers to the time dimension. On the upper part of the screen the figure shows a standard representation of a MPD with the  $y$  axis indicating the number of elements on the canvas and the line type depicting the different phase types (i.e., modeling, comprehension or reconciliation). Below the MPD, the logged events are reported. Specifically, the position of an event indicates the time of its occurrence and its color depicts the type of interaction. The bottom part of the figure, in turn, shows the AOIs the user was focusing on at a particular point in time, i.e., resulting into a depiction of the transition between AOIs over time.

The combined MPD is especially suited to obtain a better understanding of comprehension phases, i.e., phases where no tool interactions occur. The example depicted in Fig. 5 highlights two such time frames. In the first case, the user initially spent time reading the text. Then, after a quick look at the model, she concentrated on the toolbox and finally on the model again. This behavior might indicate that the user needed to understand the problem (i.e., focus on text and model) and then to find a way how to externalize it (i.e., focus on toolbox and

model). In the second case, the user flipped her gaze between text and model many times, which might indicate semantic validation.

Please note that these patterns of transitions between AOIs are used for the improvement and refinement of the automatic phases detection. Such automatic phase detection might be useful in experiments, to calculate specific metrics for phases, and also as a building block of neuro-adaptive IS [14].

### 3.2 From Static Visualizations to Movies

The first two visualizations described do not consider the time dimension: in the first case all points are plotted, one on top of the other; in the second case, if the total time spent looking at two areas is the same, it is not distinguished whether such distribution occurred at the beginning or at the end of the modeling session. To cope with this problem, we not only provide static images but *movies* reflecting the actual distribution of fixations and the distribution of the time spent at each point. In order to generate these animations, the basic idea is to concatenate frames each of them providing a snapshot of the evolution of the modeling process (in terms of model interactions and fixations). In particular, we build time frames following two different approaches: sliding window and incremental. Following a *sliding window* approach, statistics (and, therefore, the frames) are computed just for the events observed in the latest sliding window. The *incremental* approach, instead, computes the statistics and the frames with all events from the beginning until the current point.

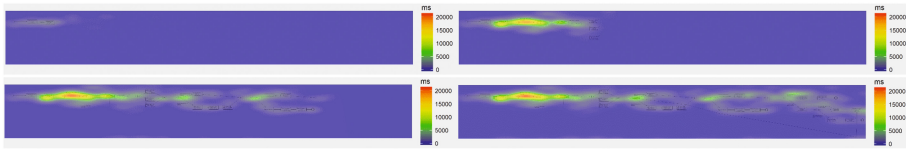
A fundamental characterization of these movies is the fact that we can depict not only the evolution of information obtained from the eye tracker but the evolution of the underlying model as well. Therefore, the first frames provide the fixations on top of an almost empty canvas, whereas the final frames report the fixations on top of the finished model. Figures 6 and 7 report few frames from movies created using the approach just mentioned. Frames reported in Fig. 6 have been created with sliding window approach (window length: 10s). Instead, frames in Fig. 7 have been created adopting the incremental policy.<sup>3</sup>



**Fig. 6.** Frames sampled from a video: evolution of the fixations distribution.

Finally, please note that there is no need for such representation for the last visualization (the Combined MDP): in this case the time dimension is explicitly reported on the x-axis of each chart.

<sup>3</sup> Animations are available at <http://bpm.q-e.at/eye-tracking-ppm>.

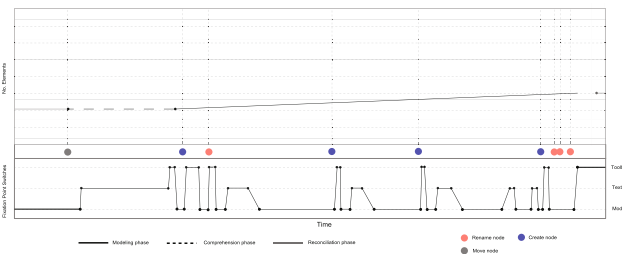


**Fig. 7.** Frames sampled from a video: evolution of the heat map over time.

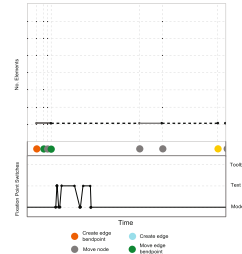
## 4 Demonstration

This section demonstrates the application of the introduced visualizations in the context of an experiment performed in 2015 where model interactions and eye tracking data of 116 subjects (all novices) were assessed, amongst other data, while creating a business process model. Details on the design, research questions and hypotheses can be found in [22]. The examples shown in this section rely on visualisations and logs of six out of the 116 subjects.

By visualizing the existing MPDs, the user interactions, and the fixation data in an integrated manner we could gain additional information about the comprehension phases and refine the fixation patterns initially proposed in [14]. For this, the fixation patterns were assessed manually for six subjects by two persons using the combined modeling phase diagram followed by a cross validation. *Problem understanding* as a pattern is characterized by exclusive fixations on the textual description (see Fig. 8): subjects were reading the textual description and building a mental model of the process. Usually, after the problem understanding pattern, subjects are focusing on the toolbox, followed by the modeling canvas before they start modeling (see Fig. 8). The participants seem to select the proper tool to externalize their mental model within that time frames. We called this pattern *method finding*. Additionally, as depicted in Fig. 9, fixation behavior alternately focusing on the textual description and the modeling canvas seems to be typical for checking if the current model corresponds to the textual description. We named that pattern *semantic validation*. The second validation pattern, also in Fig. 9, is characterized by exclusive fixations on the model. This type of validation appears to be used to check if the immanent modeling language rules are obeyed, thus, we called the described pattern *syntactic validation* as in [14].



**Fig. 8.** Problem understanding and method finding.



**Fig. 9.** Validation.

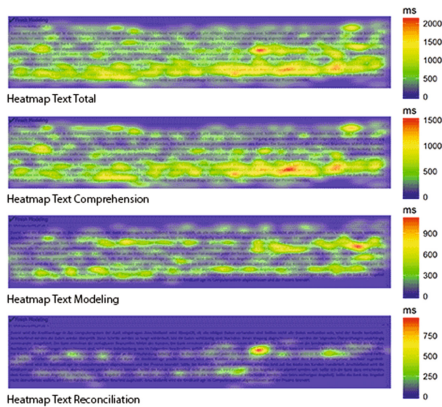


Exploring our visualizations not only provides deeper insights into comprehension phases (that were treated as black boxes in MPDs so far), but also into modeling and reconciliation phases. For example, modeling and comprehension phases alternate more often than previously thought, as depicted in Fig. 8: in this case the combined MPD shows several phases of modeling and comprehension, which were aggregated to a long modeling phase by the original classifier. This misclassification happened due to the data used by the previous classifier which did not consider eye tracking. Phases of inactivity below a threshold value were aggregated with the time frames before and after. We found similar evidence regarding reconciliation phases (Fig. 10). Here, the reconciliation phase identified by the original phase detection is interrupted by a phase of semantic validation, which was not correctly detected because the duration of this phase was below the threshold. These examples demonstrate the benefit of the combined MDP and give advice to improve phase detection algorithm to distinguish different patterns with higher accuracy. At the moment we are implementing a refined version of the automatic detection algorithm and plan to use the combined MPDs to assess the reliability of the classifier again.

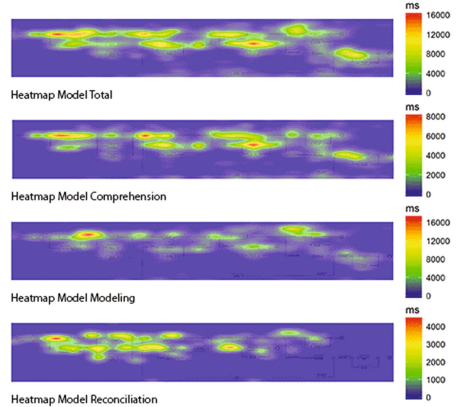


**Fig. 10.** Comprehension interrupting reconciliation.

By means of heat maps, the specific parts within the process which received most attention can be identified for every subject. Figure 11 depicts heatmap plots of the textual description for (i) the complete modeling session and additionally for the timeframes of (ii) comprehension, (iii) modeling, and (iv) reconciliation. Figure 12, in turn, is an example for the heatmap plots of the model



**Fig. 11.** Heat maps with text.



**Fig. 12.** Heat maps with model.

itself for the same timeframes as used in Fig. 11. The shown heat maps are a representative example of a single subject within the data set.<sup>4</sup> In line with [18], we expect that parts of a task which are causing difficulties will receive more attention and should thus be identifiable on a subjective level. The combined MPDs can not directly visualize the amount of attention given to a certain area of the screen while working on the task. Heat maps, in turn, are suitable to identify the most difficult parts by coloring areas depending on the total fixation durations and thus provide a very intuitive way of interpreting the given attention. Comparing the parts of highest attention on the text and the model for the total time spent on the task revealed that the difficult parts within the textual description do not necessarily correspond to the most difficult parts within the model. The parts of highest attention within the textual description deal with the entering point of a long back loop and a part describing an XOR-split and the corresponding conditions and alternative branches (Fig. 11). One of those alternative branches is the origin of the long back loop. Within the model, the XOR-splits are generally receiving more attention than the rest of the elements. But here the fact that conditions for the splits need to be defined has to be taken into account. Still, most difficult parts within the modeling canvas seem to be a loop at the beginning of the process; the entering point of the long back loop; and one ending point of the process (Fig. 12). Several subjects had problems modeling the loop at the beginning of the process, but since the subjects are novices we do not think this is representative for process modeling in general. In addition to the separation into textual description and modeling canvas we provide visualizations for the respective areas corresponding to comprehension, modeling, and reconciliation phases. For instance, as depicted in Fig. 11 “Heatmap Text Comprehension”, the attention on the text clearly focuses on the XOR-split described above, and therefore on the origin of the long back loop. The identification of the correct entering point within the text for the respective loop seems to be identified during reconciliation phases as shown in “Heatmap Text Reconciliation”. When examining the heatmap with the modeling canvas in modeling phases (Fig. 12, “Heatmap Model Modeling”), the most difficult part seems to be the small loop at the beginning of the process, pointing towards a lack of modeling knowledge once again. During phases of comprehension and reconciliation (Fig. 12, “Heatmap Model Comprehension” and “Heatmap Model Comprehension”), the origin of the long back loop received a high amount of attention, along with one ending point of the process. In sum, since the long back loop within the process occurred as difficult part within the textual description and within the model itself, we expect this part to be the most difficult overall. This fits into the statistical data: the back loop is the part with the highest error rate, with about 50% correct solutions out of 116 subjects.

To draw reliable conclusions and to use an automated PPM classifier eye tracking data of high quality is necessary. As mentioned in Sect. 3, we used fixations distribution plots to identify biases (cf. Fig. 3). In the upper corners of the screen no fixations were recorded by the eye tracker even though it is very likely

<sup>4</sup> Higher resolution versions can be found at [http://bpm.q-e.at/eye\\_tracking\\_ppm](http://bpm.q-e.at/eye_tracking_ppm).

that the subject actually fixated on those spots, since reading the whole text without skipping parts of sentences is much more natural. This deviation seems to be bend on the top corners of the screen, reminiscent of *fisheye lenses*. This bending effect seems to occur unrelated to visual aids, since the depicted participant does not wear glasses or lenses, this is also true for the opposite shaping of the bending effect we also found within the dataset. In general, eye tracking accuracy tends to be poorest in the corners of the recorded screen and is strongly dependent to particular participant's characteristics [23]. Even though the most interesting part of the screen in our case is the area around the horizontal middle line, this bending causes information loss and might be disadvantageous when using a classifier. Since the classifier is based on fixations on different AOIs, fixations falsely assigned to the modeling canvas instead of assigning it to the textual description could lead to wrong classifications. As an example, see the fixation points slightly below the textual description, within the left side of the modeling canvas in Fig. 3. Manual or automated offset compensation methods could be employed during the analysis phase, but should be avoided [19]. The described bending effect is a limitation of eye tracking technology, along with reported gaze inaccuracies up to 2 degrees which represents about 2 cm at a recording distance of 70 cm [19]. Although we do not have such general tracking inaccuracies within the bending effect free data sets, this has to be considered and shows the importance of visualization tools to assess eye tracking quality.

## 5 Conclusions and Future Work

This paper reports an innovative contribution on the inspection of the process of graphical artifact modeling and, in particular, to the modeling of business processes. Specifically, the literature describes analysis techniques focusing on the interactions with the modeling environment. In this paper we also analyze data referring to eye tracking. By combining and synchronizing these two streams, it is possible to provide visual tools which enable a deeper understanding and a more fine grained representation of users' actions and intentions. Demonstrations show that such analysis is indeed capable of providing useful insights, such as the identification of "problem understanding", and "validations" actions.

The importance of this work is based on the innovative possibilities it enables: the automatic analysis of eye tracking data, during process modeling sessions, is the main future work. Such automatic analysis will allow the extraction of detailed actions and the refinement of the MPD. In order to be able to automatically detect modeling phases, the conducted fixation patterns have to be validated. Therefore, a qualitative study with post hoc interviews would be a suitable approach. Additionally, we are going to examine modeling and reconciliation phases to clarify if the same or similar patterns can be found here. However, before starting such analysis, a deeper understanding of the behavior is needed, and the tools described in this paper represents optimal instruments.

**Acknowledgements.** This work is partially funded by the Austrian Science Fund project "The Modeling Mind: Behavior Patterns in Process Modeling" (P26609).

## References

1. Becker, J., Rosemann, M., Uthmann, C.: Guidelines of business process modeling. In: Aalst, W., Desel, J., Oberweis, A. (eds.) *Business Process Management*. LNCS, vol. 1806, pp. 30–49. Springer, Heidelberg (2000). doi:[10.1007/3-540-45594-9\\_3](https://doi.org/10.1007/3-540-45594-9_3)
2. Mendling, J.: *Metrics for Process Models: Empirical Foundations of Verification, Error Prediction, and Guidelines for Correctness*. Springer, Heidelberg (2008)
3. Mendling, J., Reijers, H.A., Recker, J.: Activity labeling in process modeling: Empirical insights and recommendations. *Inf. Syst.* **35**(4), 467–482 (2010)
4. Reijers, H.A., Mendling, J.: A study into the factors that influence the understandability of business process models. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **41**(3), 449–462 (2011)
5. Pinggera, J.: *The Process of Process Modeling*. Ph.D. thesis, Innsbruck University (2014)
6. Pinggera, J., Soffer, P., Fahland, D., Weidlich, M., Zugal, S., Weber, B., Reijers, H., Mendling, J.: Styles in business process modeling: An exploration and a model. *Softw. Syst. Model.* **14**(3), 1055–1080 (2015)
7. Claes, J., Vanderfeesten, I., Pinggera, J., Reijers, H., Weber, B., Poels, G.: A visual analysis of the process of process modeling. *Inf. Syst. e-Bus. Manage.* **13**(1), 147–190 (2015)
8. Hoppenbrouwers, S.J.B.A., Proper, H.A.E., Weide, T.P.: A fundamental view on the process of conceptual modeling. In: Delcambre, L., Kop, C., Mayr, H.C., Mylopoulos, J., Pastor, O. (eds.) *ER 2005*. LNCS, vol. 3716, pp. 128–143. Springer, Heidelberg (2005). doi:[10.1007/11568322\\_9](https://doi.org/10.1007/11568322_9)
9. Soffer, P., Kaner, M., Wand, Y.: Towards understanding the process of process modeling: Theoretical and empirical considerations. In: Daniel, F., Barkaoui, K., Dustdar, S. (eds.) *BPM 2011*. LNBIP, vol. 99, pp. 357–369. Springer, Heidelberg (2012). doi:[10.1007/978-3-642-28108-2\\_35](https://doi.org/10.1007/978-3-642-28108-2_35)
10. Martini, M., Pinggera, J., Neurauder, M., Sachse, P., Furtner, M.R., Weber, B.: The impact of working memory and the “process of process modelling” on model quality: Investigating experienced versus inexperienced modellers. *Sci Rep.* **6** (2016)
11. Pinggera, J., Zugal, S., Weidlich, M., Fahland, D., Weber, B., Mendling, J., Reijers, H.A.: Tracing the process of process modeling with modeling phase diagrams. In: Daniel, F., Barkaoui, K., Dustdar, S. (eds.) *BPM 2011*. LNBIP, vol. 99, pp. 370–382. Springer, Heidelberg (2012). doi:[10.1007/978-3-642-28108-2\\_36](https://doi.org/10.1007/978-3-642-28108-2_36)
12. Claes, J., Vanderfeesten, I., Pinggera, J., Reijers, H.A., Weber, B., Poels, G.: Visualizing the process of process modeling with PPMCharts. In: Rosa, M., Soffer, P. (eds.) *BPM 2012*. LNBIP, vol. 132, pp. 744–755. Springer, Heidelberg (2013). doi:[10.1007/978-3-642-36285-9\\_75](https://doi.org/10.1007/978-3-642-36285-9_75)
13. Blascheck, T., Kurzhals, K., Raschke, M., Burch, M., Weiskopf, D., Ertl, T.: State-of-the-Art of visualization for eye tracking data. In: *EuroVis - STARs*. The Eurographics Association (2014)
14. Weber, B., Pinggera, J., Neurauder, M., Zugal, S., Martini, M., Furtner, M., Sachse, P., Schnitzer, D.: Fixation patterns during process model creation: Initial steps toward neuro-adaptive process modeling environments. In: *HICSS*. IEEE (2016)
15. Weber, B., Neurauder, M., Pinggera, J., Zugal, S., Furtner, M., Martini, M., Sachse, P.: Measuring cognitive load during process model creation. In: Davis, F.D., Riedl, R., vom Brocke, J., Léger, P.-M., Randolph, A.B. (eds.) *Information Systems and Neuroscience*. LNISO, vol. 10, pp. 129–136. Springer, Cham (2015). doi:[10.1007/978-3-319-18702-0\\_17](https://doi.org/10.1007/978-3-319-18702-0_17)

16. Schrepfer, M., Wolf, J., Mendling, J., Reijers, H.A.: The impact of secondary notation on process model understanding. In: Persson, A., Stirna, J. (eds.) *PoEM 2009*. LNBP, vol. 39, pp. 161–175. Springer, Heidelberg (2009). doi:[10.1007/978-3-642-05352-8\\_13](https://doi.org/10.1007/978-3-642-05352-8_13)
17. Pinggera, J., Zugal, S., Weber, B.: Investigating the process of process modeling with cheetah experimental platform. In: *ER-POIS*, 13–18(2010)
18. Riedl, R., Léger, P.M.: *Fundamentals of NeuroIS*. Springer, Heidelberg (2016)
19. Holmqvist, K., Nyström, M., Andersson, R., Dewhurst, R., Jarodzka, H., Van de Weijer, J.: *Eye Tracking: A Comprehensive Guide to Methods and Measures*. Oxford University Press, Oxford (2011)
20. Ehmke, C., Wilson, S.: Identifying web usability problems from eye-tracking data. In: *British HCI, British Computer Society*, pp. 119–128 (2007)
21. Grinstein, G., Trutschl, M., Cvek, U.: High-dimensional visualizations. In: *Visual Data Mining Workshop, KDD*. Citeseer (2001)
22. Neurauter, M., Pinggera, J., Martini, M., Burattin, A., Furtner, M., Sachse, P., Weber, B.: The influence of cognitive abilities and cognitive load on business process models and their creation. In: Davis, F.D., Riedl, R., vom Brocke, J., Léger, P.-M., Randolph, A.B. (eds.) *Information Systems and Neuroscience*. LNISO, vol. 10, pp. 107–115. Springer, Cham (2015). doi:[10.1007/978-3-319-18702-0\\_14](https://doi.org/10.1007/978-3-319-18702-0_14)
23. Hornof, A.J., Halverson, T.: Cleaning up systematic error in eye-tracking data by using required fixation locations. *Behav. Res. Methods Instrum. Comput.* **34**(4), 592–604 (2002)