



A Brief of Review: Multimedia Authoring Tool Attributes

Marvin Chandra Wijaya^{1,2*}, Zulisman Maksom¹, Muhammad Haziq Lim Abdullah¹

¹Fakulti Teknologi Maklumat Dan Komunikasi, Universiti Teknikal Malaysia Melaka, Melaka 76100, Malaysia

²Department of Computer Engineering, Maranatha Christian University, Bandung 40164, Indonesia

Corresponding Author Email: marvin.cw@eng.maranatha.edu

<https://doi.org/10.18280/isi.260101>

Received: 1 November 2020

Accepted: 16 January 2021

Keywords:

multimedia authoring tool, authoring attributes, SMIL, NCL, formal verification, XML

ABSTRACT

Multimedia authoring is the process of assembling various types of media content such as audio, video, text, images, and animation into a multimedia presentation using tools. Multimedia Authoring Tool is a useful tool that helps authors to create multimedia presentations. Multimedia presentations are very widely used in various fields, such as broadcast digital information delivery, digital visual communication in smart cars, and others. The Multimedia Authoring tool attributes are the factors that determine the quality of a multimedia authoring tool. A multimedia authoring tool needs to have several attributes so that these tools can be used properly. The purpose of this literature review study is to find the advantages of the multimedia authoring tool attribute in each of the existing studies to produce knowledge on how to create a good quality multimedia authoring tool. These attributes are Editing, Services, Performance, and the Formal Verification Model. Editing attribute is an attribute for interfacing with the author. Followed by Service attribute and performance attribute to check and achieve proper multimedia documents. Since 1998, a multimedia modeling tool has been studied, and up to now, there have been many studies that have focused on one or more of these attributes. This article discusses the existing studies to examine the attributes generated from the studies. Multimedia authoring attributes are very important to study because they are the benchmarks of the software requirement specifications of Multimedia Authoring tools. The use of the Petri net model, the Hoare Logic, and the Simple Interactive Multimedia Model as a formal verification model can improve the performance of the Multimedia Authoring Tool. In the questionnaire that was submitted to the users, it was assessed positively by the users with the improvements in the Multimedia Authoring Tool.

1. INTRODUCTION

The Multimedia Authoring Tool is a tool for creating multimedia presentations to produce Multimedia documents. Multimedia authoring tools should be able to help designers to simplify and avoid errors in multimedia documents [1]. A multimedia document is a multimedia information repository containing multimedia objects and multimedia application models [2]. Most of the Multimedia Programming Languages used today are Synchronized Multimedia Integration Language (SMIL) and Nested Context Language (NCL). SMIL and NCL are part of XML which is de facto widely used for data exchange, using Data Type Definition as the modeling scheme [3]. Multimedia programming language functions for writing interactive audiovisual presentations [4]. It defines markup for timing, layout, animations, visual transitions, and other things [5]. Content from multimedia streaming can be distributed across various server locations (cloud-based) [6]. A secure framework for multimedia distribution needs to be used to ensure its security, performance, and to protect digital privacy [7-10]. A multimedia streaming needs an algorithm and a routing protocol to manage the on-demand distribution of multimedia content [11-13]. Multimedia streaming is currently widely used for various applications such as e-learning, mobile learning, telemedicine, military, and others [14-19].

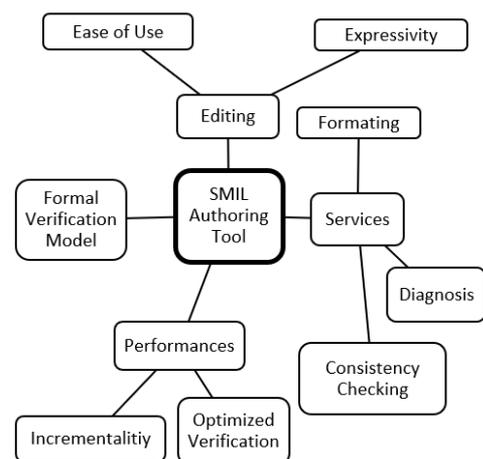


Figure 1. Multimedia authoring tool attributes

The Multimedia Authoring Tools have been created since 1998 by Dick C.A. Bulterman with the name "GriNS" (A Graphical Interface for Creating and Playing SMIL documents) [20]. The development of the multimedia authoring tool has carried out more specific research for the SMIL language as one of the multimedia programming languages. For example, the existing research on the development of SMIL multimedia

authoring tools are "A Hybrid Approach for Spatio-Temporal Validation of Declarative Multimedia Documents" Santos et al. [21] and "Steve: A hypermedia authoring tool based on the simple interactive multimedia model" by De Mattos and Muchaluat-Saade [22].

The development of Multimedia Authoring Tool research from the beginning until now has different advantages for each research. These advantages can be grouped into several categories (called Multimedia Authoring Attributes). These attributes are Editing, Services, Performances, and Formal Modeling / Verification (as in Figure 1) [23].

2. EDITING ATTRIBUTES

The Editing attribute is a facility provided to authors in creating multimedia presentations. In the process of editing a multimedia presentation, there are two factors: ease of use and expressiveness. The editing process is one of the phases in the kernel mechanism for multimedia authoring. In this phase, the interaction with the author is a matter of concern. Editing attributes in the Multimedia authoring tool include the multi-view user interface, the complexity of use, information about errors, conflicts in spatial or temporal view, syntax errors, source file errors. The use of proper and regular patterns can make application ease of use from various aspects [24]. With more and more facilities provided to the author, the value of editing attributes is getting better. The editing attributes are discussed below.

2.1 Ease of Use

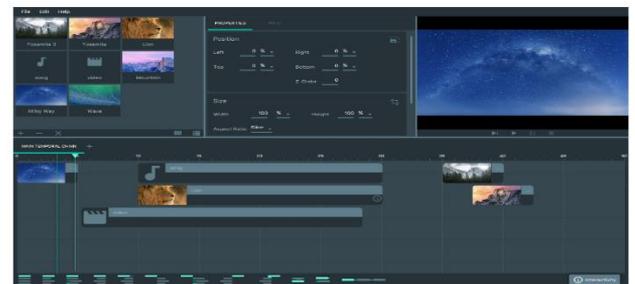
Ease of use is a factor based on the ease of the author in making multimedia presentations. One of the factors of ease of use is the "WYSIWYG" (What You See Is What You Get) authoring environment. Yang and Yang (2003) has been one of the researchers who made the user interface in the form of "WYSIWYG" [25]. In multimedia authoring tools, there are several windows: A visual layout window, a timeline window, a filter window, an attribute window, and a preview window.

In 2018, Steve conducted a study entitled "STEVE: A Hypermedia Authoring Tool based on the Simple Interactive" [22]. STEVE aims to create hypermedia applications for internet and virtual TV systems in a user-friendly way for users with no knowledge of multimedia authoring languages and templates. From the available tools, several requirements are needed to be implemented to develop user-friendly tools and allow ordinary users to create multimedia presentation documents are Temporal synchronization, Output language, Temporal view editing, Spatial View Editing, and Presentation Preview as in Figure 2a.

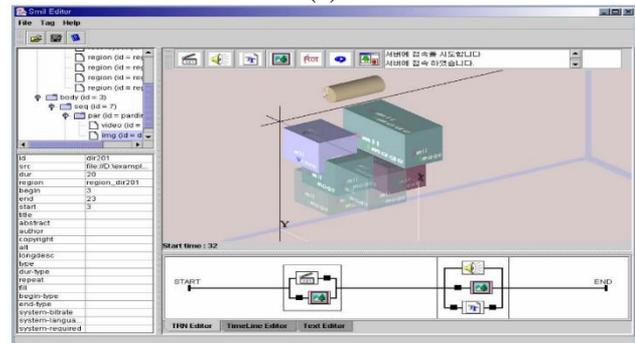
Mee Young Sung and Do Hyung Lee in 2004, conducted a study entitled "A Collaborative Authoring System for Multimedia Presentation" [26]. Mee Young Sung implemented some ideas for efficient concurrency control. The experiments conclude that the system more advantageous than traditional multimedia authoring systems in terms of authoring time and ease of interaction. In the multimedia authoring tool, the author can choose between Temporal Relations Network (TRN), timeline editor, or text editor as in Figure 2b. Temporal Relations Network is a 3-dimensional Spatio-Temporal editor (3D). In the 3D Spatio-temporal, the author can specify the temporal aspects and spatial aspects simultaneously in the form of 3D graphics. In the 3D graphic,

one axis (T zone) represents timeline information and the other two axes (XY zone) represent spatial coordinates. A visual media object (image or video) is represented in the form of a rectangular box and an audio media object is represented in the form of a cylinder.

The hierarchy in a user interface is the basis of a multimedia authoring tool [27]. The standard hierarchy of the user interface model in multimedia authoring tools is shown in Figure 3. The user interface hierarchy is a view of a multimedia presentation that consists of two main parts namely Media and Window view. The media is divided into two parts, namely non-visual media and visual media. Non-visual media is audio. Visual media consists of two parts, namely continuous and non-continuous media. Continuous media are video and animation. Non-continuous media are images and text. The View editor consists of a spatial view, temporal view, text view/editor, and message zone for interaction with the author. Top-down and bottom-up strategies can be used to help novice authors to create multimedia documents [28].



(a)



(b)

Figure 2. (a) Example of Multimedia Authoring User Interface [26]; (b) Temporal Relation Network [27]

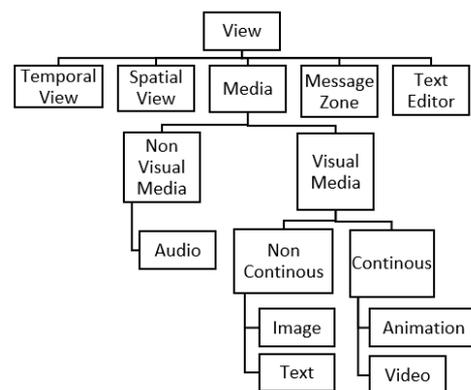


Figure 3. Hierarchy of Multimedia Authoring tools user interface

2.2 Expressivity

The characteristics of multimedia authoring in a multimedia authoring tool are very important for the author. Characteristic factors in expressivity multimedia authoring are document logical structures, representation of media objects, presentation characteristic specifications separated from a component definition, reuse, definition of links in a composition domain, complex relationships among components, flexibility in temporal specifications, and adaptation to a presentation platform [29].

- Document logical structures

Document logical structuring is a form of a grouping of elements that can be used to create specifications for a multimedia presentation. The composition of the grouping elements is a facility to present groups in parallel or sequentially in making multimedia presentations.

- Representation of media objects

A multimedia authoring that has media groupings based on the basic types of media such as text, audio, image, and video. This facility allows a definition of a media object that is independent of media content. This means that media content can be used by several different objects.

- Presentation characteristic specifications separated from a component definition

A multimedia authoring can also have the facility to provide specifications for multimedia presentation components on different objects. This makes it possible to create several different multimedia presentation specifications with the same components or vice versa to make a multimedia presentation with different media components.

- Reuse

An important feature of multimedia authoring tools is that they allow the reuse of a media object in another document or other part of a document. Through object's references to avoid changing an object specification more than once will make the authoring process simpler.

- Definition of links in a composition domain

Another facility is to allow a media object to have a different link on reuse. For example, in the first composition media object A has two links to media objects B and C. In the second composition, media object A has a different link, namely in media object D.

- Complex relationships among components

An expressive multimedia authoring can define complex relationships between media objects. For example, if media object A and media object B have finished being presented, then media object C begins to be presented. Another example is if media object A is being presented and media object B is finished being presented then media object C is starting to be presented.

- Flexibility in temporal specifications

Another facility is the ability to specify the temporal behavior of a media object flexibly. Temporal specifications are for defining temporal intervals such as duration, time begin, and time end. Another flexibility is to make adjustments to improve presentation time if unexpected events occur.

- Adaptation to a presentation platform

Another facility is to adapt a multimedia presentation to the platform on which the multimedia is presented. These parameters include user profile information, language used or quality of service information, and bandwidth.

The study entitled "SMIL Author: An Authoring System for SMIL-Based Multimedia Presentations" provides copy, cut, and paste facilities on the application toolbar to strengthen the expressivity attributes [30].

3. SERVICES ATTRIBUTES

The method used in this research is to use six generic steps in conducting a review analysis. These steps are as follows:

- Perform formulations in search of questions or objectives.
- Search for further literature materials
- Screening for inclusion
- Doing preliminary studies
- Perform data extraction
- Perform data analysis

This research method is suitable for the problem of this study. Problem-solving in this study uses secondary data that can be solved using a six generic step method in conducting a review analysis.

The specific implementation for the objective formulation process is as follows:

- Clarify the basic concepts involved in this research process.
- Doing the right research question selection
- Establish a suitable research question

The implementation of a literature review search in this study is to match existing secondary data in existing research. The data sought is related to the Multimedia Authoring Tools. In a preliminary study, the process of compiling a multimedia authoring hierarchy is carried out as shown in Figure 3. The search results for literature that match preliminary studies will be followed by data extraction. Data extraction was collected following the objectives of this study. The final step in the research method in this study is to analyze the data from the results of data extraction in the previous step.

Services are measured by the services provided by Multimedia Authoring tools to the authors. The services provided are divided into three parts: Formatting, Consistency Checking, and Diagnosis. The service attributes are discussed below.

3.1 Temporal formatting

Formatting attributes are emphasized on the temporal formatting in creating multimedia presentations. Formatting attributes can be assessed if the multimedia authoring tool developed has a temporal view editing or timeline view editing. Temporal formatting is pre-verification for temporal conflict. While the author creates a temporal view or timeline view, the authoring tool does the initial verification if the author makes a mistake in inserting a multimedia presentation scenario in the temporal view. Formatting algorithms need to take into account non-deterministic environments and need to take into account sudden improvements when the presentation is running. Problems that arise during multimedia presentations are network delay, user interaction [31]. HyperProp is a system that represents the Spatio-temporal structure in a multimedia presentation. Using the hyper Prop System can correct errors in the temporal formatting attribute. Hyper Prop System performs real-time consistency checking on elastic time [31].

3.2 Consistency checking

Consistent checking is a temporal checking of all phases in multimedia authoring (Visual Editor, Kernel Mechanism, and SMIL documents). Consistency checking is important in multimedia authoring and multimedia documents because it ensures the multimedia aspects of the presentation can be executed [23, 32]. Creating a multimedia presentation involves writing temporal and spatial relations, writing constraints on temporal writing is sometimes inconsistent. The inconsistencies must be identified as soon as possible and the inconsistencies are eliminated. Consistency and inconsistency checks can be processed on the XML metadata used in multimedia authoring [33]. Eliminating inconsistencies can save time and simplify the algorithmic modeling process of a multimedia presentation [34]. The interface hierarchy as shown in Figure 3 will be processed by the Semantic Validator Module shown in Figure 4. Multimedia presentations also need to do semantic checking (semantic verification), As shown in Figure 4a and Figure 4b, semantic checking is processed in two phases (editor phase and playback phase) by Annalisa Bossi and Ombretta Gaggi [35].

The Semantic Validator Module (Figure 4a) will have the following effects:

- The Semantic Validator Module can set the initial duration value of a media in multimedia presentations.
- Semantic Validator Module can check nested tags in SMIL documents.
- The Semantic Validator Module can check conflicting values on the designed multimedia presentation.
- The author can check the consistency of the SMIL language used in the designed multimedia presentation.

The Semantic Validator Module was modified and developed by Gaggi (Figure 4b) to provide additional functions. The Semantic Validator Module can be implemented in various authoring interfaces. This modification also has an automatic pipeline function to validate many SMIL documents.

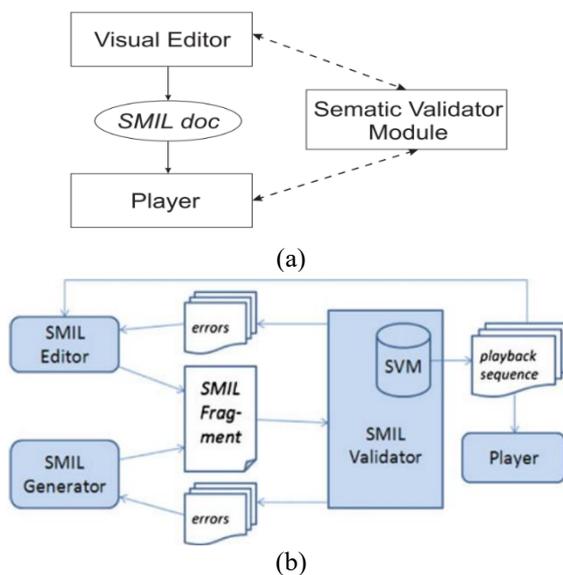


Figure 4. (a) Semantic Validator; (b) SMIL Validator by Gaggi [35]

In 2010, Jack Jansen, Pablo Cesar, Dick C. A. Bulterman from CWI: Centrum Wiskunde & Informatica, Amsterdam conducted a study entitled “A Model for Editing Operations on Active Temporal Multimedia Documents” [36]. Jansen conducted a study for the categorization of modification operations. These categories make consistency in temporal semantics. These categories also become a different implementation for temporal semantics. Verification that is processed by the editing operation model on active temporal multimedia is pseudo-classes as follows:

- Invalid operation: There is an invalid operation that causes invalid multimedia documents. An example is the incorrect attribute change on a media object.
- Trivial operation: Operation on the declarative part of <head> element that has no direct effect.
- Out of scope operations: Operations of media objects that fail semantic-checking verification.
- Side effect: Checking for indirect effects occurs when the attribute changes on a media object.

3.3 Diagnosis service

Diagnosis Service is a Multimedia Authoring tool that can provide error information that occurs to the author. Error information given to the author is in the form of line location information in the multimedia programming code that experiences errors or conflicts. The information provided also contains error information that occurred at the location of the line. Usually, the user interface has a special window or pop window to provide information to the author. As shown in Figure 5a, a Messages Zone in user interfaces in the multimedia authoring tool for service diagnosis [23]. As shown in Figure 5b, an Error Zone Message in a user interface in the multimedia authoring tool for service diagnosis [37].

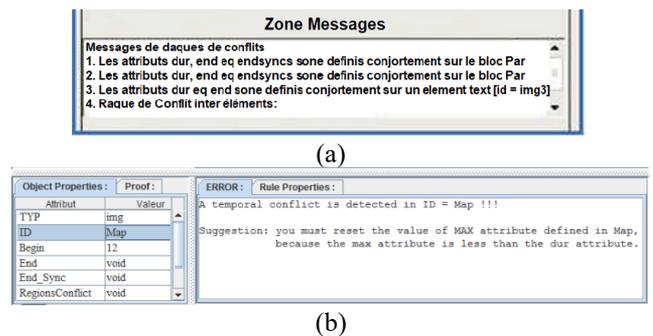


Figure 5. (a) Zone messages for diagnosis service; (b) Error message for diagnosis service

4. PERFORMANCE ATTRIBUTES

The Performance Attribute provides an overview of the system from the Multimedia Authoring tool. The measured factor is incremental feedback when creating Multimedia programming language coding or when documenting specifications. The performance attributes are discussed below.

4.1 Incrementality

Incrementality is the ability to process data that considers changes in input data [38]. The concept of Incremental is the ability of the software to recalculate only the changed input

data. This will save processing time [39]. To create a system that can detect errors as quickly as possible during the editing process, a system with a model that has incremental verification is needed. Modeling called the SMIL Net hierarchy (H SMIL Net) can carry out incremental verification. Systems that have the ability to incremental verification have the advantage of having a good response time Graphical representation for body place, sequence place, and parallel place are shown in Figure 6 [32].



Figure 6. Graphical representation of composite places

4.2 Optimized verification

Verification is an important thing in Multimedia Authoring besides modeling, verification, and modeling are two things that are inseparable in the Kernel Mechanism. Optimized verification has several ways such as whole verification, local verification. Verification can be in the form of temporal verification or spatial verification in a Multimedia document [40].

Verification can be in the form of Multimedia Programming code or in the form of temporal view or timeline view. Verification is done on document specifications and platform specifications, media properties that are not intrinsic (can be semantic checking), and temporal inconsistencies. There are several verification strategies, including Reachability Analysis, Time Computation, Model Checking, and Rule Validation [39].

4.2.1 Time computation

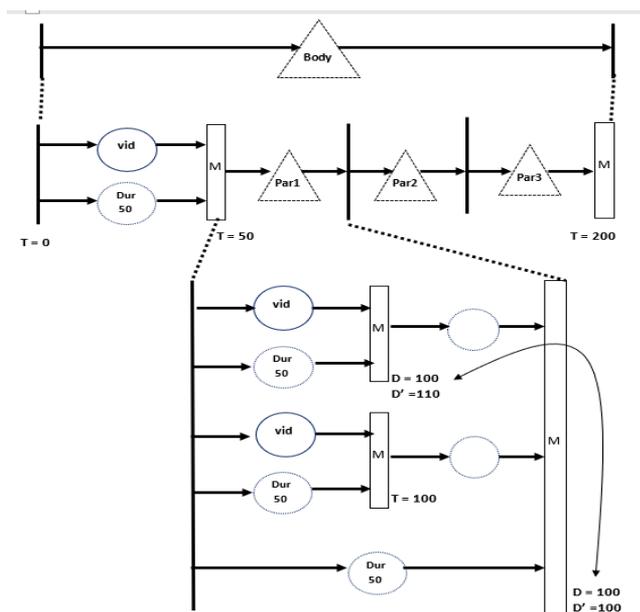


Figure 7. Example Time conflict detection

In the time computation strategy, calculating the overall duration of time from a Petri Net Model hierarchy, every time there is a change the calculation is done in stages from the local level first. If at the local level there are no errors, then the

calculation will proceed to a higher level. But if it turns out there is an error at a level/group of elements/media, the calculation and verification process will be stopped first and give an error warning to the author. An example of temporal verification processes using time computation with the H-SMIL Net model is shown in Figure 7.

If the duration of the "vid" element is changed from 50 to 60, the algorithm for detecting the change are:

- Step1: $D(\text{par1}) = 100$ and $D'(\text{par1}) = 110$
- Step 2: $D(\text{par1}) = D(\text{par1})$
- Step 3: "par 1" is invariant

where:

D = Total duration before modification

D' = Total duration after modification

It was found that the value of $D = 100$ which is invariant (because it is explicitly written in a multimedia document) turns out to have a different value from the results of the calculation of time. This causes the Kernel Mechanism to detect errors and stop the process further.

4.2.2 Spatiotemporal conflict verification

In 2012, Samia Bouyakoub and Abedkader Belkhir conducted a study entitled A Spatio-Temporal Authoring Tool for Multimedia SMIL document [40]. This study discusses spatial conflicts that will reduce the quality of multimedia presentations. This happens because there are some overlapping objects. In non-visual objects such as audio, overlapping occurs in time. In visual objects such as video, image, and text, overlapping occurs in time and layout. As shown in Figure 8, there are two types of spatial conflicts, the first is the entire spatial conflict in two regions (region R1 and region R2). The second is partial spatial conflict in two regions (region R1 and region R2). Spatial conflicts are allowed if playing time is different, but if the playing time is the same then this conflict does not occur. Spatial conflicts that occur during playing at the same time are called Spatio-temporal conflicts.



Figure 8. Example entire spatial and partial spatial Conflict between two region R1 and R2

There are 169 possible spatial relations in the two regions, that Bouyakoub did for grouping overlapping spatial cases. Of the 169 cases, there were 81 cases of overlapping spatial. Baoyakuob divides into 5 case classes by considering the following properties as shown in Figure 9.

Class 1: $(Y2 < Y1 < Y2+H2) \ \&\& \ ((X1 < X2+L2) \ \&\& \ ((X1+L1) > X2))$

Figure 9a are forms of spatial conflict that occur in overlapping conflict class 1. There are 19 forms that are

classified as overlapping class 1.

Class 2: $(Y2 < Y1+H1 < Y2+H2) \ \&\& \ ((X1 < X2+L2) \ \&\& \ (X1+L1) > X2)$

Figure 9b are forms of spatial conflict that occur in overlapping conflict class 2. There are 14 forms that are classified as overlapping class 2.

Class 3: $((Y2 < Y1+H1 < Y2+H2) \ \&\& \ (Y2 \leq Y1)) \ \&\& \ ((X1 < X2+L2) \ \&\& \ (X1+L1 > X2))$

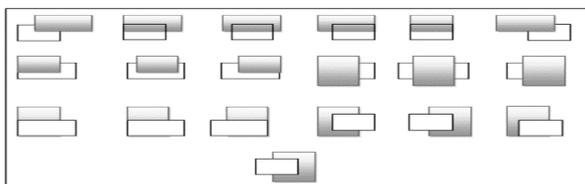
Figure 9c are forms of spatial conflict that occur in overlapping conflict class 3. There are 10 forms that are classified as overlapping class 3.

Class 4: $((Y1 \leq Y2) \ \&\& \ (Y2+H2 \leq Y1+H1)) \ \&\& \ ((X2 < X1 < X2+L2))$

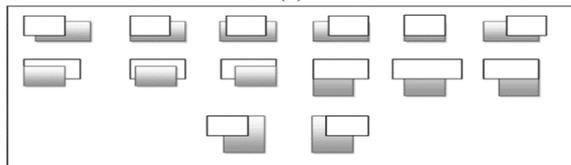
Figure 9d are forms of spatial conflict that occur in overlapping conflict class 4. There are 7 forms that are classified as overlapping class 4.

Class 5: $((Y1 \leq Y2) \ \&\& \ (Y1+H1 \geq Y2+H2)) \ \&\& \ ((X1 \leq X2) \ \&\& \ (X1+L1 \geq X2+L2)) \ || \ ((Y2 \leq Y1) \ \&\& \ (Y2+H2 \geq Y1+H1)) \ \&\& \ ((X2 \leq X1) \ \&\& \ (X2+L2 \geq X1+L1))$

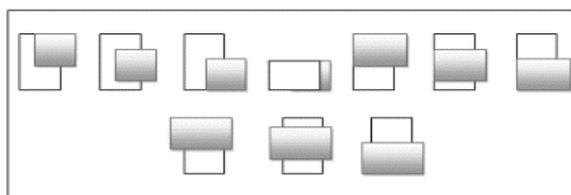
Figure 9e are forms of spatial conflict that occur in overlapping conflict class 4. There are 26 forms that are classified as overlapping class 5.



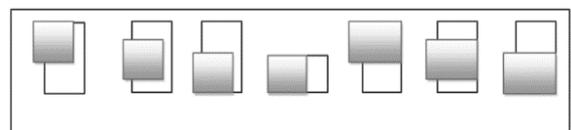
(a)



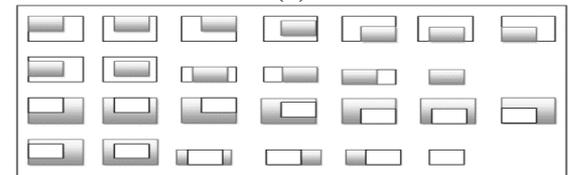
(b)



(c)



(d)



(e)

Figure 9. Spatial conflict of (a) case 1 (b) case 2 (c) case 3 (d) case 4 (e) case 5

Spatial verification is detecting overlapping spatial at certain time intervals and looking for empty areas that will not

cause spatial conflicts. As an example illustrated in the program the following causes:

```

<smil>
<head>
<layout>
  <root-layout width="800" height="800"/>
  <region id="reg1" width="400" height="300" top="0"
left="100"/>
  <region id="reg2" width="400" height="300"
top="500" left="0"/>
  <region id="reg3" width="200" height="200"
top="600" left="600"/>
</layout>
</head>
<body>
  <par>
    <video src="vid1.mp4" id="img1" region="reg1"
dur="50"/>
    <video src="vid2.mp4" id="img1" region="reg2"
dur="50"/>
    <video src="vid3.mp4" id="img1" region="reg3"
dur="50"/>
  </par>
</body>
</smil>

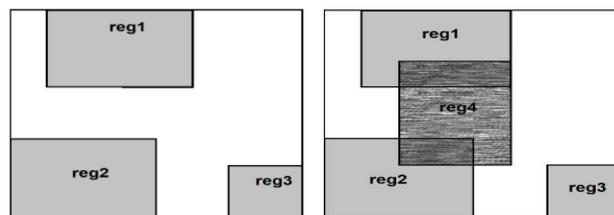
```

Examples of the programs will result in the position of the media as in the Figure 10a. For example, the author will add an element that is played in the parallel group with the previous program as follows:

```

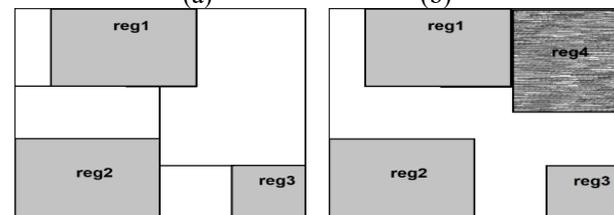
<region id="reg4" width="300" height="400" top="200"
left="200"/>
<video src="vid3.mp4" id="img1" region="reg4"
dur="50"/>

```



(a)

(b)



(c)

(d)

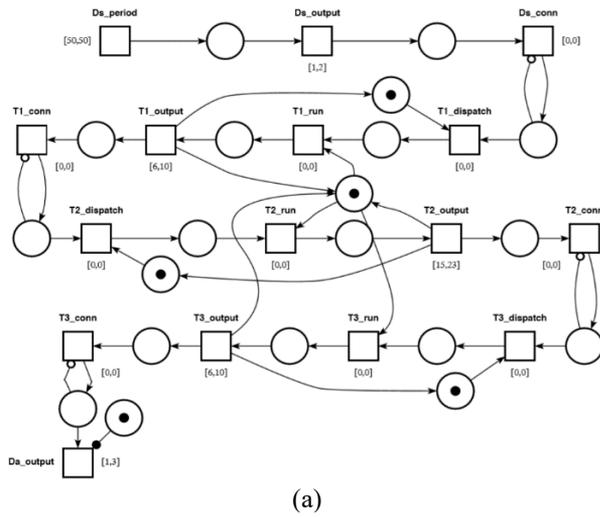
Figure 10. (a) Spatial view form example program (b) Spatial Object overlapping (c) Area calculation for Spatial conflict (d) Spatial conflict correction: region reg4 displays on another area

Figure 10b is an example of spatial overlapping that occurs on object D with two objects (Object A and Object B). The Spatio-temporal authoring system will function to correct the conflict or overlapping spatial. The system will start searching for areas that are still empty. The system will calculate the area

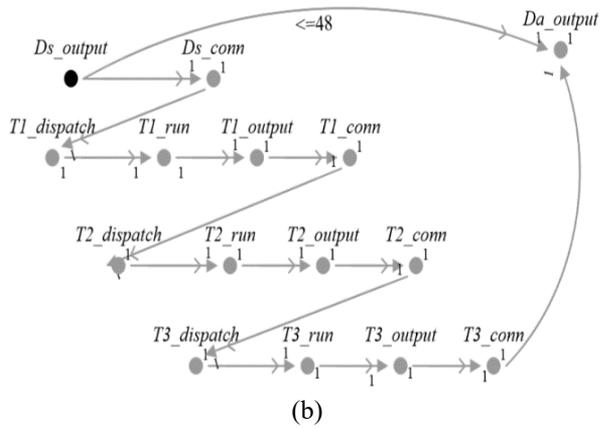
in a rectangle according to the area that is filled in the reg1, reg2, and reg3 regions (Figure 10c). After the system finds an area that can be used for the reg4 region, the system will propose an area that can be used as shown in Figure 10d. Indiscriminate spatial changes in the information on a video can reduce the quality of the video reference. This means that spatial changes must be carried out correctly so that quality is maintained [41, 42]. Video quality can be measured so that the quality can be maintained, one of the methods is the video motion quality (VMQ) algorithm [43].

4.2.3 Model checking

Time Petri Net Analyzer or called TINA (<http://www.laas.fr/tina>) is a software for editing and analyzing Petri Nets and Time Petri Nets (TPN). TINA can be used to find out conflicts such as deadlocks, linear temporal properties, or bisimilarity. TINA can be given input form in textual format (in XML) and graphical [44].



(a)



(b)

Figure 11. (a) Example of a scenario using TPN [45] (b) A scenario of the flow latency from the example [45]

The Model checking on Time Petri Nets is checking the trip token from the initial place to reaching the destination place. Then check whether the token remains at the destination place sometime later. Checking can be processed using recorded behavior, which is called conformance checking [46]. For example, searching for tokens can be seen in Figure 11a which has 19 places and 14 transitions. Starting from the token which is in the initial place until reaching the “Da_output” transition. The results of the search can be described as in Figure 11b, by looking at the scenario of the flow latency [45].

In 2016, Fatma Zohra Mekalia, Abdelghani Ghomari, Samy Yazid, and Djamel Djenouri conducted a study entitled “Temporal and Spatial Coherence Verification in SMIL Documents with Hoare Logic and Disjunctive Constraints: A Hybrid Formal Method” [37]. The study discusses the problem of conducting formal verification on a SMIL document. Modified Hoare logic is used to find this solution. If there is a conflict or overlapping spatially and temporally, then there is a system of reporting error conditions. The reporting is useful for the author to fix the existing conflict.

A SMIL file that will be evaluated, will enter the spatial and temporal verification system. If there are spatial and temporal conflicts, there are messages and options to correct the error. The author can choose the repair scenario options offered by the system. The verification system starts with checking the temporal attribute and spatial attribute of each media object as following:

Media play at same time

$$: \left\{ \begin{array}{l} ((start_m \geq start_n) \wedge \\ (start_m \leq start_n + dur_n)) \vee \\ (start_n \geq start_m) \wedge \\ (start_n \leq start_m + dur_m)) \end{array} \right\} \quad (1)$$

Media takes the same place

$$: \left\{ \begin{array}{l} (left_n \leq left_m + height_m) \wedge \\ ((top_m width_m) \leq (top_m - width_n)) \wedge \\ ((top_n - width_m) \leq top_m) \end{array} \right\} \quad (2)$$

4.3 Experimental, test, and evaluation

A Multimedia Authoring Tool can be evaluated and tested in two ways. The first evaluation is to check the correctness of the output of the tool. Checks were carried out using the World Wide Web SMIL Test Suite [47].

The Test Suite uses linking and interaction tests on multimedia documents. The test scenario uses the following parameters:

- General link
- Selection of start/end on each multimedia element
- Link-activated start / end execution
- Selection of priority for multimedia elements
- Use of <excl> activation
- A temporal search of links

Table 1. SMIL player testing

| Players | Temporal linking | Start/ end tags | Event Base <excl> |
|----------------------|------------------|-----------------|-------------------|
| Internet Explorer | No Support | Support | Support |
| Real Player | Support | Support | Support (Poor) |
| Ambulant SMIL Player | Support | Support | Support |

The results of the SMIL player evaluation experiments are shown in Table 1 and Figure 12. The Internet explorer player successfully performs the basic functions of a SMIL player, the player can execute start and end tags properly. Internet Explorer does not support temporal linking, the experiment always fails all temporal linking tests. On the <excl> event activation attempt, the Internet Explorer player managed to run the tags properly. The Real Player has successfully performed the basic functions of a SMIL Player by executing start and

end tags. Real Player also managed to execute the base <excl> event tag properly, but while checking temporal linking, it encountered several failures. Ambulant SMIL Player has successfully performed the basic functions of a SMIL Player by executing start and end tags. Ambulant Player also successfully executed the base <excl> event tag properly. At the time of checking temporal linking, the Ambulant Player experienced a tag execution failure on the temporal linking.

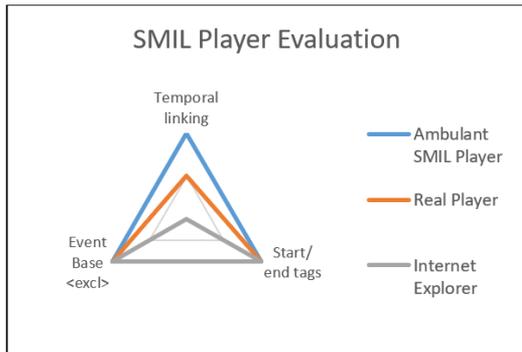


Figure 12. SMIL player evaluation

A Multimedia Authoring Tool is tested by creating a questionnaire [35]. The questionnaire was submitted to ten users. The questionnaire contains four questions for users to measure the usefulness of the Multimedia Authoring tool. Users answer questions by rating on a scale of 1 to 5. The four questions are as follows:

- Q1: How useful is the Multimedia Authoring Tool?
- Q2: How many errors can the Multimedia Authoring Tool detect?
- Q3: How is the user experience using Multimedia authoring Tool?
- Q4: How do the instructions for using the Multimedia Authoring Tool relate to the attempt to learn them?

Table 2. Questionnaire results

| Question | Answers | | | | | Total of Users |
|----------|---------|---|---|---|---|----------------|
| | 1 | 2 | 3 | 4 | 5 | |
| Q1 | | 1 | 4 | 4 | 1 | 10 |
| Q2 | 1 | 4 | 4 | 1 | | 10 |
| Q3 | | | 4 | 6 | | 10 |
| Q4 | | | 4 | 5 | 1 | 10 |

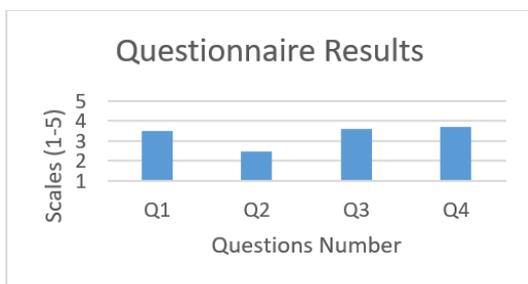


Figure 13. Questionnaire results

The results of the questionnaire are shown in Table 2 and Figure 13. In the first question that discusses the usefulness of the Multimedia Authoring Tools from user perceptions, the average scale value is 3.5. It can be seen that users rate positively on the usefulness of the Multimedia Authoring Tool. In the second question, which discusses the many errors

detected by Multimedia Authoring tools, the average scale value is 2.5. It appears that there is still a need for continuous improvement on the Multimedia Authoring Tool to be able to detect and validate errors that may occur when making multimedia presentations. In the third question, which discusses user experience using Multimedia Authoring Tools, the average scale score is 3.6. This shows that the user has a positive perception while using the tool. In the third question which discusses the ease of use of Multimedia Authoring Tools, the average scale value is 3.7. This shows that the user has a positive perception that the effort to use the tool is not difficult.

5. FORMAL VERIFICATION MODEL ATTRIBUTES

Formal verification is a process to prove a system has the correctness of the system's functions. Formal verification is very useful to prove the correctness of a system in software. Formal verification uses formal models or abstractions from mathematical models.

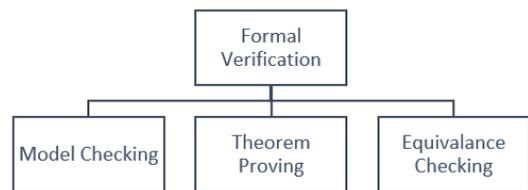


Figure 14. Formal verification techniques

Formal verification techniques use 3 ways to perform verification, namely Model Checking, Theorem Proving, and Equivalence Checking (as shown in Figure 14).

Model Checking is a state-based formal verification approach. Simple Interactive Multimedia Model can be used by the Multimedia Authoring Tool to check the model. Theorem Proving is a process of verifying that a system implementation is under the specifications of the software. Hoare Logic can be used by the Multimedia Authoring Tool to perform Theorem Proving. Equivalence Checking is a process to verify the functions of software have accuracy. Petri Net can be used by a Multimedia Authoring Tool to verify equivalence.

A Petri Net has the advantage of being able to process the flow of multimedia presentations. A Petri net is a group of directed arcs that connects places and transitions in a multimedia presentation. Places stores tokens [48]. Marking a Petri Net is assigning tokens to places. Modified Petri nets can be modified for several specific purposes. Real-Time Synchronization Model (RTSM) is a modification of the Petri net to represent the temporal layout of multimedia presentations [49]. The Petri Net model with a non-deterministic event is a modification of the Petri net to divide the temporal layout into several parts for easier processing [25]. Hierarchy SMIL Petri Net (H-SMIL Net) is a modification of the Petri net to be able to verify the temporal and hyper-temporal consistency [32, 40, 48]. Time Petri Nets (TPN) is a modification of Petri nets to verify temporal consistency for NCL multimedia programming language [50]. Petri Net is not only processed normally but also can be processed and combined with fuzzy logic [51, 52]. Petri Net with Fuzzy Logic is a modification of the Petri net to perform a Spatio-temporal knowledge base using fuzzy logic [53, 54].

Another multimedia modeling document is Hoare Logic.

Hoare Logic is a formal system for verifying a computer program using logic rules [35]. Hoare Logic Rules Enrichment is a modification of Hoare Logic that aims to verify temporal inconsistencies in SMIL documents [37]. Hoare Logic uses mathematical equations to perform the theorem verification process on a system in multimedia authoring tools. Hoare Logic has the advantage of fully verifying a system for multimedia presentations.

Simple Interactive Multimedia Model or called SIMM is a multimedia document model that aims to use event-based temporal synchronization. The Simple Interactive Multimedia Model is an event-based model that can occur synchronously or asynchronously [22]. SIMM uses states that can model a multimedia presentation system. The modeling has the advantage of increasing the success of error checking in Multimedia Authoring tools.

6. SUMMARY

Table 3 shows the research that is used as data for reviewing

Table 3. Summary

| Attribute | Sub Attribute | Paper |
|---------------------------|------------------------|--------------------------------------|
| Editing | Ease of Use | [22][25][26][27][28] |
| | Expressivity | [29][30] |
| Service | Formatting | [31][31] |
| | Consistency Checking | [23][32][34][35][36] |
| | Diagnosis | [23][37] |
| Performance | Incrementality | [32][38][39] |
| | Optimized Verification | [37][39][40][44][46][45] |
| Formal Verification Model | | [22][32][35][37][40][48][49][53][54] |

REFERENCES

- [1] Picinin, D., Farines, J.M., Santos, C.A.S., Koliver, C. (2018). A design-oriented method to build correct hypermedia documents. *Multimedia Tools and Applications*, 77(16): 21003-21032. <https://doi.org/10.1007/s11042-017-5325-2>
- [2] Belhadad, Y., Refoufi, A., Roose, P. (2018). Spatial reasoning about multimedia document for a profile based adaptation: Combining distances, directions and topologies. *Multimedia Tools and Applications*, 77(23): 30437-30474. <https://doi.org/10.1007/s11042-018-6080-8>
- [3] Haw, S.C., Soong, E. (2020). Performance evaluation on structural mapping choices for data-centric XML documents. *Indonesian Journal of Electrical Engineering and Computer Science*, 18(3): 1539-1550. <https://doi.org/10.11591/ijeecs.v18.i3.pp1539-1550>
- [4] Bulterman, D.C.A. (2018). SMIL: Synchronized Multimedia Integration Language. *MediaSync*, pp. 359-385. https://doi.org/doi.org/10.1007/978-3-319-65840-7_13
- [5] Hoshka, P. (1998). An introduction to the synchronized multimedia integration language. *IEEE Multimedia*, 5(4): 84-88. <https://doi.org/10.1109/93.735872>
- [6] Banerjee, S., Chakraborty, C., Paul, S. (2019). Programming Paradigm and the Internet of Things. *Handbook of IoT and Big Data*, pp. 145-164.
- [7] Bindu, G.H., Anuradha, C., Chandra Murthy, P.S.R. (2018). A survey on multimedia content protection mechanisms. *International Journal of Electrical and Computer Engineering (IJECE)*, 8(6): 4204. <https://doi.org/10.11591/ijece.v8i6.pp4204-4211>
- [8] Edan, N., Abdullah, E.Y. (2020). Design and implementation of a novel secured and wide WebRTC signalling mechanism for multimedia over internet. *International Journal of Electrical and Computer Engineering*, 10(5): 5430-5435. <https://doi.org/10.11591/IJECE.V10I5.PP5430-5435>
- [9] Prajapati, Y.N., Srivastava, M.K. (2019). Novel algorithms for protective digital privacy. *IAES International Journal of Robotics and Automation*, 8(3): 184-188. <https://doi.org/10.11591/ijra.v8i3.pp184-188>
- [10] Yghoubi, Y.J.I.A., Pang, W.L., Wong, S.K., Chan, K.Y. (2019). Performance evaluation of video streaming on LTE with coexistence of Wi-Fi signal. *Bulletin of Electrical Engineering and Informatics*, 8(3): 890-897. <https://doi.org/10.11591/eei.v8i3.1580>
- [11] Khalifa, O.O., Ahmed, D.E.M., Hashim, A.H.A., Yagoub, M. (2019). Video streaming over Ad Hoc on-demand distance vector routing protocol. *Bulletin of Electrical Engineering and Informatics*, 8(3): 863-874. <https://doi.org/10.11591/eei.v8i3.1510>
- [12] Alomari, S.A., Alzboon, M.S., Al-Batah, M.S., Zaqibeh, B. (2020). A novel adaptive schema to facilitates playback switching technique for video delivery in dense LTE cellular heterogeneous network environments. *International Journal of Electrical and Computer Engineering*, 10(5): 5347-5367. <https://doi.org/10.11591/IJECE.V10I5.PP5347-5367>
- [13] Syamsuddin, I., Nur, R., Nirwana, H., Abduh, I., Al-Dabass, D. (2017). Decision making analysis of video

- streaming algorithm for private cloud computing infrastructure. *International Journal of Electrical and Computer Engineering*, 7(6): 3529-3535. <https://doi.org/10.11591/ijece.v7i6.pp3529-3535>
- [14] Mohammed Jameel, S., Croock, M.S. (2020). Mobile learning architecture using fog computing and adaptive data streaming. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 18(5): 2454. <https://doi.org/10.12928/telkomnika.v18i5.16712>
- [15] Yousafzai, A., Chang, V., Gani, A., Noor, R.M. (2016). Multimedia augmented m-learning: Issues, trends and open challenges. *International Journal of Information Management*, 36(5): 784-792. <https://doi.org/10.1016/j.ijinfomgt.2016.05.010>
- [16] Wijaya, M.C. (2019). Dampak Penggunaan Media Pembelajaran Berbasis Multimedia Interaktif pada Komunikasi Pengajar dan Siswa di Bandung. *Journal Pekommas*, 4(1): 53-60. <https://doi.org/10.30818/jpkm.2019.2040106>
- [17] Hakim, F.N., Solechan, A. (2018). Design and implementation multimedia learning success for vocational schools. *International Journal of Electrical and Computer Engineering*, 8(2): 1067-1073. <https://doi.org/10.11591/ijece.v8i2.pp1067-1073>
- [18] Seta, H.B., Wati, T., Muliawati, A., Hidayanto, A.N. (2018). E-learning success model: An extension of delone & mclean is' success model. *Indonesian Journal of Electrical Engineering and Informatics*, 6(3): 281-291. <https://doi.org/10.11591/ijece.v6i3.505>
- [19] Chiramdasu, R. (2019). Extended statistical analysis on multimedia concealed data detections. *Ingenierie des Systemes d'Information*, 24(2): 161-165. <https://doi.org/10.18280/isi.240205>
- [20] Bulterman, D.C.A. (1998). GRiNS: A GRaphical INterface for creating and playing SMIL documents. *Computer Networks*, 30(1-7): 519-529. [https://doi.org/10.1016/S0169-7552\(98\)00128-7](https://doi.org/10.1016/S0169-7552(98)00128-7)
- [21] Santos, J.A.F. Dos, Muchaluat, D.C., Rosin, C., Layda, N. (2018). A Hybrid Approach for Spatio-Temporal Validation of Declarative Multimedia Documents. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 14(4): 1-24. <https://doi.org/10.1145/3267127>
- [22] De Mattos, D.P., Muchaluat-Saade, D.C. (2018). Steve: A hypermedia authoring tool based on the simple interactive multimedia model. *Proceedings of the ACM Symposium on Document Engineering 2018*, 14(4): 1-24. <https://doi.org/10.1145/3209280.3209521>
- [23] Bouyakoub, S., Belkhir, A. (2011). SMIL builder: An incremental authoring tool for SMIL documents. *ACM Transactions on Multimedia Computing, Communications and Applications*, 7(1): 1-30. <https://doi.org/10.1145/1870121.1870123>
- [24] Pulungan, F.F., Sudiharto, D.W., Brotoharsono, T. (2019). Pattern lock and the app based on context, ease of use aspect in comparison. *Indonesian Journal of Electrical Engineering and Informatics*, 7(1): 58-66. <https://doi.org/10.11591/ijece.v7i1.936>
- [25] Yang, C.C., Chu, C.K., Wang, Y.C. (2008). Extension of Timeline-based Editing for Non-deterministic. *Journal of Information Science and Engineering*, 24(5): 1377-1395.
- [26] Sung, M.Y., Lee, D.H. (2004). A collaborative authoring system for multimedia presentation. In 2004 IEEE International Conference on Communications (IEEE Cat. No. 04CH37577), 3: 1396-1400. <https://doi.org/10.1109/ICC.2004.1312741>
- [27] Linaje, M., Preciado, J.C., Rodriguez-Echeverria, R., Conejero, J.M., Sánchez-Figueroa, F. (2017). An smil-timesheets based temporal behavior model for the visual development of web user interfaces. *Journal of Web Engineering*, 16(7-8): 371-394.
- [28] Akiyama, Y., Oore, S., Watters, C. (2015). Framework for constructing task-space to support novice multimedia authoring. *Multimedia Tools and Applications*, 74(15): 6111-6147. <https://doi.org/10.1007/s11042-014-1911-8>
- [29] Antonacci, M.J., Muchaluat-Saade, D.C., Rodrigues, R.F., Soares, L.F.G. (2020). Improving the expressiveness of XML-based hypermedia authoring languages. In *Multimedia Modeling: Modeling Multimedia Information and Systems, MMM 2000*, pp. 71-88. https://doi.org/10.1142/9789812791993_0006
- [30] Yang, C.C., Yang, Y.Z. (2003). SMILAuthor: An authoring system for smil-based multimedia presentations. *Multimedia Tools and Applications*, 21(3): 243-260. <https://doi.org/10.1023/A:1025770817293>
- [31] Soares, L.F.G., Rodrigues, R.F., Muchaluat Saade, D.C. (2000). Modeling, authoring and formatting hypermedia documents in the HyperProp system. *Multimedia Systems*, 8(2): 118-134. <https://doi.org/10.1007/s005300050155>
- [32] Bouyakoub, S., Belkhir, A. (2008). H-SMIL-Net: A hierarchical Petri net model for SMIL documents. In *Tenth International Conference on Computer Modeling and Simulation, Uksim*, pp. 106-111. <https://doi.org/10.1109/UKSIM.2008.54>
- [33] Kaliappan, V., Ali, N.M. (2018). Improving consistency of UML diagrams and its implementation using reverse engineering approach. *Bulletin of Electrical Engineering and Informatics*, 7(4): 665-672. <https://doi.org/10.11591/eei.v7i4.1358>
- [34] Elias, S., Easwarakumar, K.S., Chbeir, R. (2006). Dynamic consistency checking for temporal and spatial relation in multimedia presentations. In *Proceedings of the ACM Symposium on Applied Computing*, 2: 1380-1384. <https://doi.org/10.1145/1141277.1141596>
- [35] Gaggi, O., Bossi, A. (2011). Analysis and verification of SMIL documents. *Multimedia Systems*, 17(6): 487-506. <https://doi.org/10.1007/s00530-011-0233-1>
- [36] Jansen, J., Cesar, P., Bulterman, D.C.A. (2010). A model for editing operations on active temporal multimedia documents. In *Proceedings of the 10th ACM symposium on Document Engineering*, pp. 87-96. <https://doi.org/10.1145/1860559.1860579>
- [37] Mekahlia, F.Z., Ghomari, A., Yazid, S., Djenouri, D. (2017). Temporal and spatial coherence verification in smil documents with hoare logic and disjunctive constraints: A hybrid formal method. *Journal of Integrated Design and Process Science*, 20(3): 39-70. <https://doi.org/10.3233/jid-2016-0020>
- [38] Carlsson, M. (2002). Monads for incremental computing. *ACM SIGPLAN Notices*, 37(9): 26-35. <https://doi.org/10.1145/583852.581482>
- [39] Júnior, D.P., Farines, J.M., Koliver, C. (2012). An approach to verify live NCL applications. In *WebMedia'12 - Proceedings of the 2012 Brazilian Symposium on Multimedia and the Web*, pp. 223-231. <https://doi.org/10.1145/2382636.2382685>
- [40] Bouyakoub, S., Belkhir, A. (2012). A Spatio-Temporal

- authoring tool for multimedia SMIL documents. *International Journal of Multimedia Technologies*, 2(3): 83-88.
- [41] Rahman, F.D.A., Agrafiotis, D., Khalifa, O.O., Zhang, F. (2018). Reduced-Reference video quality metric using spatial information in Salient Regions. *Telkomnika (Telecommunication Computing Electronics and Control)*, 16(3): 965-973. <https://doi.org/10.12928/TELKOMNIKA.v16i3.9036>
- [42] Rahman, F.D.A., Ibrahim, A.I., Agrafiotis, D. (2018). Reduced-reference video Quality metric using spatio-temporal activity information. *Telkomnika (Telecommunication Computing Electronics and Control)*, 16(2): 909-914. <https://doi.org/10.12928/TELKOMNIKA.v15i4.9044>
- [43] Alsrehin, N.O., Klaib, A.F. (2019). VMQ: An algorithm for measuring the video motion quality. *Bulletin of Electrical Engineering and Informatics*, 8(1): 231-238. <https://doi.org/10.11591/eei.v8i1.1418>
- [44] Berthomieu, B., Vernadat, F. (2006). Time Petri nets analysis with TINA. In *Third International Conference on the Quantitative Evaluation of Systems, QEST 2006*, pp. 123-124., <https://doi.org/10.1109/QEST.2006.56>
- [45] Monteverde, D., Olivero, A., Yovine, S., Braberman, V. (2008). VTS-based specification and verification of behavioral properties of AADL models. In *International Workshop on Model Based Architecting and Construction of Embedded Systems (ACES'08)*. Springer-Verlag.
- [46] Mukhlash, I., Rumana, W.N., Adzkiya, D., Sarno, R. (2018). Business process improvement of production systems using coloured petri nets. *Bulletin of Electrical Engineering and Informatics*, 7(1): 102-112. <https://doi.org/10.11591/eei.v7i1.845>
- [47] Bulterman, D.C.A. (2004). A linking and interaction evaluation test set for SMIL. In *Proceedings of the ACM Conference on Hypertext*, pp. 46-47. <https://doi.org/10.1145/1012807.1012825>
- [48] Belkhir, A., Bouyakoub-Smail, S. (2007). Formal design of smil documents. In *Webist 2007 - 3rd International Conference on Web Information Systems and Technologies, Proceedings*, pp. 396-399., <https://doi.org/10.5220/0001271403960399>
- [49] Yang, C., Huang, J. (1996). A multimedia synchronization its implementation in transport. *IEEE Journal on Selected Areas in Communication*, 14(1): 212-225. <https://doi.org/10.1109/49.481706>
- [50] Yovine, S., Olivero, A., Córdoba, L. (2010). An approach for the verification of the temporal consistency of NCL applications. In *Anais Estendidos do XVI Simpósio Brasileiro de Sistemas Multimídia e Web*, pp. 179-184.
- [51] Xu, X.G., Xiong, Y., Xu, D.H., Liu, H.C. (2020). Bipolar fuzzy Petri nets for knowledge representation and acquisition considering non-cooperative behaviors. *International Journal of Machine Learning and Cybernetics*, 11(10): 2297-2311. <https://doi.org/10.1007/s13042-020-01118-2>
- [52] Shi, H., Wang, L., Li, X.Y., Liu, H.C. (2020). A novel method for failure mode and effects analysis using fuzzy evidential reasoning and fuzzy Petri nets. *Journal of Ambient Intelligence and Humanized Computing*, 11(6): 2381-2395. <https://doi.org/10.1007/s12652-019-01262-w>
- [53] Ribaric, S., Hrkac, T. (2012). A model of fuzzy spatio-temporal knowledge representation and reasoning based on high-level Petri nets. *Information Systems*, 37(3): 238-256. <https://doi.org/10.1016/j.is.2011.09.010>
- [54] Coelho, A.L.V., Ricarte, I.L.M. (2004). On the fuzzy spatio-temporal specification of multimedia synchronisation scenarios. *Applications and Science in Soft Computing*, pp. 53-60. ISBN: 978-3-540-40856-7.
- [55] Monita, V., Irawati, I.D., Tulloh, R. (2020). Comparison of routing protocol performance on multimedia services on software defined network. *Bulletin of Electrical Engineering and Informatics*, 9(4): 1612-1619. <https://doi.org/10.11591/eei.v9i4.2389>