A Probabilistic, Ontological Framework for Safeguarding the Intangible Cultural Heritage

GIANNIS CHANTAS, SOTIRIS KARAVARSAMIS, SPIROS NIKOLOPOULOS, and IOANNIS KOMPATSIARIS, Information Technologies Institute, Centre of Research and Technology Hellas

In this article, we propose Multi-Entity Bayesian Networks (MEBNs) as the probabilistic ontological framework for the analysis of the Tsamiko and Salsa dances. More specifically, our analysis has the objective of the dancer assessment with respect to both choreography execution accuracy and the synchronization of the dance movements with the musical rhythm. For this task, we make use of the explicit, expert-provided knowledge on dance movements and their relations to the musical beat. Due to the complexity of this knowledge, the MEBNs were used as the probabilistic ontological framework in which the knowledge is formalized. The reason we opt for MEBNs for this task is that they combine Bayesian and formal (first-order) logic into a single model. In this way, the Bayesian probabilistic part of MEBNs was used to capture, using example data and training, the implicit part of the expert knowledge about dances, i.e., this part of the knowledge that cannot be formalized and explicitly defined accurately enough, while the logical maintains the explicit knowledge representation in the same way ontologies do. Moreover, we present in detail the MEBN models we built for Tsamiko and Salsa, using expert-provided explicit knowledge. Last, we conduct experiments that demonstrate the effectiveness of the proposed MEBN-based methodology we employ to achieve our analysis objectives. The results of the experiments demonstrate the superiority of MEBNs to conventional models, such as BNs, in terms of the dancer assessment accuracy.

CCS Concepts: • **Mathematics of computing → Expectation maximization**; • **Computing methodologies → Probabilistic reasoning**; *Description logics*; *Ontology engineering*; Temporal reasoning; • **Applied computing → Performing arts**;

Additional Key Words and Phrases: Intangible cultural heritage, multi-entity bayesian networks, multimodal semantic analysis

ACM Reference format:
Giannis Chantas, Sotiris Karavarsamis, Spiros Nikolopoulos, and Ioannis Kompatsiaris. 2018. A Probabilistic, Ontological Framework for Safeguarding the Intangible Cultural Heritage. *J. Comput. Cult. Herit.* 11, 3, Article 12 (August 2018), 29 pages.
https://doi.org/10.1145/3131610

1 INTRODUCTION

In the cultural heritage (CH) domain, multimedia analysis has been extensively used in the past de cades for safeguarding and treasuring intangible cultural heritage (ICH) activities and practices (Vecco 2010), via, for

The research leading to these results has received funding from the European Community's 7th Framework Programme (FP7-ICT-2011-9) under grant agreement no FP7-ICT-600676, "i-Treasures: Intangible Treasures - Capturing the Intangible Cultural Heritage and Learning the Rare Know-How of Living Human Treasures."
Authors' addresses: G. Chantas, S. Karavarsamis, S. Nikolopoulos, and I. Kompatsiaris, Information Technologies Institute, Centre of Research & Technology - Hellas, 6th km Harilaou - Thermi, 57001, Thessaloniki, Greece; emails: gchantas@iti.gr, s.karavarsamis@gmail.com, {nikolo, ikom}@iti.gr.

https://doi.org/10.1145/3131610

example, automatic indexing for convenient access and training new ICH practitioners and students. The analysis of multimedia content is a necessity that grows fast these days considering the popularity of digitizing cultural content for safeguarding both tangible and intangible resources. When it comes to ICH, the task of semantic analysis becomes even more challenging and complex than the tangible case (Vecco 2010), since the significance of intangible heritage artifacts is implied in their context, and the scope of the preservation extends also to the preservation of the background knowledge that puts these artifacts in proper perspective. These intangible assets may, for instance, be derived from performing arts (e.g., singing and dancing). Thus, semantic multimedia analysis is essential for mapping the low-level features originating from the signal of the utilized sensors (e.g., sound, image, and Electroencephalography (EEG)) to important aspects that define the examined art (e.g., singing or dancing style).

Semantic analysis, applied on multimedia content (e.g., audiovisual), aims at imitating the cognitive human ability of detecting and recognizing entities, e.g., objects, persons, sounds, or even more abstract concepts, such as relations between entities and activities that are manifested in the content. In general, semantic analysis is achieved either through trained models or models that explicitly represent the *domain knowledge*. In the latter case, the knowledge is explicitly provided by human experts. In the former case, knowledge is implicit and it is captured through training the model using example data. Last, there are semantic analysis methods that combine the characteristics and thus the benefits of both aforementioned approaches, such as the approach proposed in this work. More specifically, we aim to perform analysis of dance performances based on information in a more abstract level than plain dancer trajectories. We note, however, that while we deal with the analysis of multimedia content, our aim is to solve a problem that could be proven very useful when dealing with the more general problem of ICH contextualization and documentation.

One of the most prominent class of models that falls into the category of explicit knowledge representation are ontologies (Chandrasekaran et al. 1999). They use description logic, a variant of the first-order logic (FOL), to perform reasoning based on the modeled knowledge. However, since ontologies rely on FOL, they can only express knowledge under absolute certainty, which is a serious drawback when there is uncertainty in this knowledge. Thus, other models are needed to overcome this limitation, i.e., to model both kinds of knowledge. In general, there are two types of uncertainty, epistemic and stochastic (Aloulou et al. 2015). The first type is due to, for example, the dancer's subjective way of following the predetermined dance movements, and the latter due to the inaccuracy of any step classifier used for recognizing the dancer's moves. In any case, all uncertainty sources can be modeled efficiently in a common probabilistic framework, in which the degree of uncertainty is expressed by probability.

In this work, we propose Multi-Entity Bayesian Networks (MEBNs) (Laskey 2008; Park et al. 2013) as a model alternative to ontologies to overcome their aforementioned shortcomings when we particularly deal with the problem of semantically analyzing ICH audiovisual content. More specifically, we deal with the analysis of Greek Tsamiko (GreekBoston 2017) and Salsa dance performances and particularly with the assessment of the dancer with respect to his/her ability to properly execute the choreography in a synchronized manner. This methodology takes as input the attribute values of the elementary dance concepts (i.e., *medium-level* features that are extracted in a pre-processing stage, along with their temporal information). The elementary concepts found in Tsamiko and Salsa dance performances are essentially the body movements of the dancer (i.e., steps) and the beats of the accompanying music. Then, abstract concepts belonging in a higher level of the feature semantic hierarchy are detected in the performances, such as the dance style, to further be used for the assessment task. In other words, we exploit the inherent characteristic of the dance, which is that the dance movements occur according to a rhythmical pattern that is the same with the music accompanying the dance. Last, note that, although our purpose is not to solve the general problem of contextualization and documentation in every different ICH domain, the style information, extracted as a by-product of the assessment method, can be used in an ICH preservation environment, where sophisticated storing, indexing, and information presentation is necessary.

From a Bayesian modeling perspective, MEBNs are first-order BNs (FOBNs) (Laskey 2008; Park et al. 2013), since they can construct BNs with a customized structure depending on the situation. From the knowledge representation perspective, MEBNs are a probabilistic, ontological framework that, because of the BN construction ability, falls into the Knowledge Base Model Construction (KBMC) category (Braz et al. 2008). Thus, we opt for MEBNs to combine the advantages of Bayesian networks (BNs) (Pearl 2003) and FOL (Russell and Norvig 1995; Sowa 1999), since MEBNS are capable of representing explicit domain knowledge, through FOL, and modeling implicit and uncertain knowledge through BNs. The fact that MEBNs rely on FOL renders them a natural extension of ontologies. This means one can use an existing ontology and built a MEBN as its probabilistic counterpart. Last, with this work, we extend our previous works found in Chantas et al. (2014a, 2014b). The main large difference with them is that the they deal with a much simpler, unimodal problem, while the present problem is *multimodal*.

As stated previously, the analysis in MEBNs goes through a custom Bayesian network, constructed for the specific situation, i.e., a situation specific BN (SSBN). Among others, the construction entails the assignment of specific values to the local conditional probabilities residing at the SSBN nodes. Since these values may not be known (e.g., provided by the expert knowledge), as in the present case, we derive the Expectation Maximization (EM) (Bishop 2006; Dempster et al. 1977) algorithm, designed and employed for MEBN training. The use of EM has been favored due to its inherent ability in handling missing (incomplete) training data. By "incomplete," we refer to data that have missing values, or, in other words, we do not know the values for some variables. Moreover, our contribution comes to complement the very limited amount of works for MEBN training. The most representative example of such a work is Park et al. (2013), where the authors use an algorithm for learning both the structure and the parameters of an arbitrary MEBN model by utilizing *complete* (i.e., without missing values) training data. However, the herein proposed EM algorithm can work with incomplete data (i.e., some data have missing values), in contrast with the work in Park et al. (2013).

The rest of this document is organized as follows. In Section 2, we present the related work. In Section 3, we first define the multimodal aspect of the problem that we have to solve and then provide the MEBN-based formulation of our semantic analysis problem. In Section 4, we derive the EM algorithm for learning the MEBN parameters. In Section 5, experiments demonstrate the efficiency of the MEBN-based methodology in respect to the analysis of the dance performances. In the last section, conclusions and directions for future work are discussed.

2 RELATED WORK

As already mentioned, the proposed approach combines implicit and explicit knowledge towards analysing the multimedia content. In this respect, it can be considered to relate with different aspects dealing with domain knowledge representation, multi-modal analysis, and analysis through Bayesian inference and first-order logic. In the following, we review the existing literature in relation to each of these aspects.

2.1 ICH Domain Knowledge Representation

Ontologies (Chandrasekaran et al. 1999) act as knowledge representation models that have been successfully applied to numerous domains of knowledge. This is mainly because they provide a convenient and expressive formal language, i.e., (variants of) description logic (DL) (Baader 2003), which practically shares with FOL the same expressive power (Sowa 1999). Indeed, a very important application of ontologies for safeguarding the ICH are platforms that, via indexing and categorization along with efficient searching and content presentation tools, manage to aggregate content of various CH domains in a large-scale database and provide convenient access to that data (Staab and Studer 2013a). Thus, ontologies have been applied to a vast and diverse range of domains, including CH (Staab and Studer 2013a). A notable example is the CIDOC Conceptual Reference Model (CRM) (Doerr 2003) implemented as an extensible ontology for representing heterogeneous CH domain knowledge, and works using this model, as, for example, those in Hu et al. (2014), Le Boeuf (2012), Kettula and Hyvönen

(2012), and Tan et al. (2009). Another important example falling into the knowledge representation category is the Europeana data model (EDM) (Europeana 2017). This model, which was developed for the implementation of the Europeana digital library (Europeana 2017), was designed with the purpose to enforce interoperability between various content providers and the library. Also, an interesting work undertaken in the context of the ECLAP project is that described in Bellini and Nesi (2015). In this article, the European Collected Library of Artistic Performance (ECLAP) semantic model is presented, which is specifically defined for aggregating ICH content coming from heterogeneous providers. ECLAP has the role of content aggregation for Europeana. The article also presents the mapping of the ECLAP model to Europeana Data Model (EDM). Last, it is worthwhile to refer the reader to the comparative study found in Hug and Gonzalez-Perez (2012) of knowledge representation models for the domain of cultural heritage.

Also, there are narrower domain specific ontologies, as, for example, that in El Raheb and Ioannidis (2011) and Aristidou et al. (2015), which encodes traditional dances (the former encodes Tsamiko among others) by making use of the Laban motion analysis principles. Other works that use Laban motion analysis for folk dances are Aristidou and Chrysanthou (2013), which focuses on emotion recognition through dancing, and Kapadia et al. (2013), which achieves fast and efficient indexing and retrieval of human motion captured based in an efficient mapping from key sequences to motions. Another motion indexing and retrieval methodology that is based on semantic analysis is that presented in Müller et al. (2005), where the temporal and spatial features are extracted to this end.

An ontology for representing musical knowledge, where the focus is on the representation of chord sequences, is introduced in Wissmann (2012). In Mallik et al. (2011), an ontology is proposed for the preservation of Indian traditional dances. As a last example, an ontology framework is proposed in Lombardo et al. (2016) for safeguarding and enabling convenient access to drama, which is widely considered an ICH practice. Tangible cultural heritage ontologies also exist, such as that in Isemann and Ahmad (2014), which is used for facilitating the access to a digital repository of three-dimensional (3D) cultural heritage items, that in Grieser et al. (2011) are used for museum exhibits. The Benesh movement notation has also been used, as, for example, in De Beul et al. (2012), to semantically annotate human motion in videos. To this end, an ontology model and semantic concept classifiers are employed, based on this notation and using Ontology Web Language (OWL) (Staab and Studer 2013b).

However, standard ontologies have a reasoning capability confined by the expressiveness limitations of DL on which they are based. This is due to the incapability of DL to express uncertain knowledge, since it is grounded in a deterministic, logical framework. For example, as stated in the study (Chen and Nugent 2009), ontologies should be extended so as to be applicable to domains with inherent uncertainty, which holds also for our case.

There were many efforts towards a consistent probabilistic ontological framework that combines deterministic with probabilistic modeling tools, see the survey in Braz et al. (2008). One major category mentioned in Braz et al. (2008) is the Knowledge Based Model Construction (KBMC) category, to which MEBNs belong, where the standard ontology-based reasoning is replaced with (Bayesian) inference. This inference is applied on a custom probabilistic (Bayesian) model that is constructed on demand. Notably, although all models of this category share many similarities, MEBNs are recognized as the richest of the KBMC approaches. This was a major motivation for us to opt for MEBNs as a model that we employ to face the challenges posed in this work.

Apart from KBMC-based approaches, another attempt to utilize BNs in such a way is the object-oriented BN (OOBN) (Koller and Pfeffer 1997) that allows the encoding of the modeled entities by treating them as objects (in the same spirit with object-oriented programming languages). Moreover, Probabilistic Relational Models (PRMs) (Pfeffer and Koller 2000) extend relational knowledge bases (i.e., databases) to model probabilistic relations. As in the KBMC approach, in PRMs, a BN that expresses the joint probability distribution over these random variables is constructed. For a more extended discussion about the similarities and differences of these models with MEBNs, see Laskey (2008). So far, to the best of our knowledge, MEBN remains the richest model available for our modeling and analysis goals.

2.2 Multimodal Analysis

Next, we refer to works analyzing ICH (dance) performances with a focus mostly on works that follow an analysis in a multimodal fashion. In Drémeau and Essid (2013), a probabilistic multimodal method for aligning two dances, i.e., an expert and a learner performance, is proposed to correct the learner by indicating the deviations of his/her performance with respect to that of the expert. In Naveda and Leman (2008), a method for the analysis of body movement in Samba dances using information of the musical meter is proposed. More specifically, a multi-modal approach is adopted that recognizes spatial patterns existing in repetitive movement of different moving body parts. To achieve that, a search for periodicities is applied, steered by the knowledge of the musical meter. Last, there is a multimodal corpus of Salsa dance recordings described in Essid et al. (2013). Using this dataset, in Essid and Richard (2012), Alexiadis and Daras (2014), and (Essid et al. 2012), Salsa dance analysis methods are presented that classify the Salsa steps. Shiratori et al. (2006) present a work that achieves to provide an animation of a human that dances in synchronization with the rhythmic of an arbitrary input music by modeling efficiently the relation of musical rhythm with that of human motion. Also, in Shapiro et al. (2006), a method that, although not multimodal, is closely related to the human motion analysis performed in our work is presented that identifies human motion styles and, using this information, can alter the motion style of an animated human avatar. Last, in Chan et al. (2011), virtual reality system with automated dancer scoring is presented, which based on motion tracking and matching.

For some of the most well-known and ubiquitously used probabilistic models, applied for multimodal analysis, see the survey in Bahador et al. (2013). One of the most well-studied and applied probabilistic types of model is the BN (Pearl 2003). More specifically, BNs are acyclic and directed probabilistic graphical models that achieve a balanced tradeoff between inference in a reasonable time and sufficient modeling complexity. For this reason, BNs have been applied successfully to many problems of diverse fields, as well as to multimodal analysis problems, since they can naturally model multimodal data.

An important work that employs BNs in the context of the motion analysis and human action recognition (HAR), which is based the Laban motion annotating system, is proposed in Rett et al. (2010). This method uses BNs to model the relationship between low-level motion features and Laban-based motion concepts (i.e., mid-level features). Bayesian inference using this BN achieves the analysis of the motion and the inference of the action taken by the subject of the analysis. However, there is main drawback in this method, which is that the Bayesian network is designed once and kept fixed like this in every situation.

Dynamic Bayesian networks (Ghahramani 1998) extend Hidden Markov Models (HMMs) by unrolling a BN of arbitrary number of nodes and structure instead of a two-node BN, as in HMMs. Thus, a DBN comes with a BN construction mechanism that uses (a) a standard Bayesian Network as a template for building the whole DBN just by concatenating copies of it repeatedly to make a chain of the BN copies, (b) a standard BN as the first element of the chain, and (c) a set of BN arcs for connecting the nodes of the nodes of two subsequent BN copies. HMMs (Rabiner and Juang 1986) are also well-known models that are can be seen as a special case of DBNs, since they have a two node BN as a repetitive basic element (Rabiner and Juang 1986).

We have to note at this point that the structure of BNs either of a standard BN or the BN that unrolls in HMMs and DBNs is designed *a priori* either by an expert user or an automated procedure. Examples of the latter approach, i.e., automatic learning of the BN structure, can be found in Park et al. (2013). Thus, HMMs and DBNs, although drastically augmented, still suffer from serious modeling limitations, which is the fixed structure of the unrolling BN. However, probabilistic models falling into the KBMC category, such as MEBNs, are flexible enough to adjust their structure according to the needs of virtually every possible situation.

3 RHYTHM-BASED MULTIMODAL ANALYSIS USING MEBNS

In this section, we present our work in applying MEBNs to the problem of the analysis of Tsamiko and Salsa dances. Initially, we present the multimodal aspect of the problem we aim to tackle, the FOL theory for modeling

Table 1. Description of the Multimodal Nature

Time steps			1	2	3	4	5	6	7	8	9	10	
Virtual	Rhythm	Event	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9	R_{10}	…
		Event Class	Rclass1	Rclass2	Rclass3	Rclass4	Rclass5	Rclass6	Rclass7	Rclass8	Rclass9	Rclass10	…
Physical modalities	A	Event no.	A_1		A_2		A_3		A_4		A_5		
		Event Class	Aclass1		Aclass2		Aclass3		Aclass1		Aclass2		
	B	Event no.	B_1			B_2			B_3			B_4	
		Event Class	Bclass1			Bclass2			Bclass3			Bclass1	
	C	Event Class	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9		
		Event Class	Cclass1	Cclass2	Cclass3	Cclass4	Cclass5	Cclass6	Cclass1	Cclass2	Cclass3		

The name of an event (e.g., Step, Beat) is produced by the concatenation of the modality name and an increment

the synchronization between different modalities. Finally, we describe the MEBN models that are based on this theory to facilitate the analysis of Tsamiko and Salsa dances, respectively. For more information about the theory and mechanism behind MEBNs, see Laskey (2008).

3.1 Dealing with Multimodality

In our multimodal analysis framework, there are two modalities in the Tsamiko and Salsa recordings that we have at our disposal: the visual in which the basic body movements are manifested and the audio in which the music beats are manifested. Before the main analysis, we apply a pre-processing step, which entails the extraction of the information about the prominent events occurring in each modality; see Section 5 for more details.

Next, we describe how we deal with multimodality for a general case. Suppose that we have N modalities, as depicted in Table 1, named A, B, C, . . . , and so on. In this table, the horizontal line denotes the time axis. This implies that the closer the projection of two events to this axis, the smallest their temporal difference is. Also, each event belongs to a class and all events of the same class have the same feature values. Thus, the fact that the event class, for a specific modality, recurs periodically means that the event features recur also with this period.

Based on the above, one can see that the key idea here is the abstract concept of rhythm, which can be seen as a periodic process during which the R_i events occur in regular intervals where their class recurs periodically. From a more technical point of view, by applying the above framework in the present problem, we can model the rhythmic structure of Tsamiko and Salsa. More specifically, our model evaluates the synchronization between two modalities by checking how accurately they correlate with rhythm events rather than directly with each other. Indeed, knowing the rhythmic structure and the frequency of the rhythm events, it is feasible to define implicitly the correct timing of an event by indicating which rhythm event(s) should be close in time, as well as the degree of closeness. This approach allow us to tackle the problem at the level of aligning events rather than aligning whole sequences with each other, something that facilitates the employment of expert knowledge in our MEBN-based analysis task. Moreover, since this is done in a probabilistic setting, we manage to also model the degree of uncertainty in our measurements of the event timings.

3.2 First-Order Bayesian Logic

FOBL is the probabilistic logic that plays the role of the theoretical foundation of MEBNs. FOBL extends FOL with the ability to model with probabilistic tools the inherent uncertainty existing in the knowledge of the attribute values, which knowledge can be either come from measurements or expert knowledge. In FOL, the basic logical elements are the functions and predicates. Predicates are in essence functions that return either the "true" or "false" constant; for this reason, when using the "function" term in what follows, we will refer to both functions and predicates. Last, apart from functions, in FOL there are logical rules that define which constants the functions are *valid* to return, for every possible input, which can be constants and/or other functions. Moreover, FOBL retains these deterministic elements of FOL, while in parallel uses probabilistic extensions of them. Based on both the deterministic and probabilistic elements, in FOBL analysis is performed by generating an SSBN.

Next, we describe the core mechanism of MEBNs that generates an SSBN using FOBL. First, there are blueprints that create the r.v.'s of an SSBN according to the needs of an arbitrary modeling situation that arises; these are a key element of MEBNs. These "blueprints" are called lifted random variables (LRVs), because they are *abstract* random variables that act as templates to produce regular random variables. Also, there are probabilistic rules (i.e., extended logical rules), which define dependencies between LRVs. More specifically, they define the dependencies of the regular random variables produced by the LRVs that created them. In FOBL, there can be both logical functions and lifted random variables as well as logical and probabilistic rules. In the next two paragraphs, the above modeling elements are formally defined and explained in detail.

Logical functions and rules: A function is defined as follows:

$$\mathrm{func}_a(n_1, n_2, \dots), \tag{1}$$

where the input arguments enclosed by the parentheses are logical variables that take constant values. Also the function returns a constant that depends on the input. The logical rules (see next) define which returned constants are valid given the input, while there can be only one such a constant. An example of a logical rule that defines a valid constant of the above function given other functions is the following:

$$\mathrm{func}_a(n_1, n_2, \dots) = \mathrm{Const}_a \leftarrow \mathrm{func}_b(k_1, k_2, \dots) = \mathrm{Const}_b, \mathrm{func}_c(m_1, m_2, \dots) = \mathrm{Const}_c, \dots . \tag{2}$$

An example of defining the returned constant of a function is the following:

$$\mathrm{func}_a(\mathrm{Const}_1, \mathrm{Const}_2, \dots) = \mathrm{Const}_a, \tag{3}$$

which must comply with the rule given in Equation (2). In the above example, the returned constant of func_a (i.e., Const_a) is deterministically defined by the constants Const_b, $\mathrm{Const}_c, \dots$. In other words, the left arrow denotes that the constants of the right-hand functions imply the constant of the left-hand term. Last, it is important to explain the arguments that the functions take. These are logical variables that can be assigned with (or else be instantiated to) a constant value, which value belongs in a specific set. This set is normally predefined by other logical rules that, in essence, declare which values are valid to replace the logical variables. Also, it is worthwhile to mention that, in the above example, it is implied that the rule holds for all valid logical variable replacements (instantiations). The same holds also for the logical rules in the rest of the article.

Lifted random variables and probabilistic rules: A lifted random variable (LRV) is a "blueprint" used to create regular r.v.'s. An LRV returns a constant *randomly* chosen from a finite set of constants and with a specific probability. Also, the probabilistic rules define lifted probabilistic dependencies between the lifted random variables. This means that the regular random variables produced by the lifted ones have dependencies with each other as defined by these rules. Thus, a probabilistic rule is in essence an extension of a logical rule. To explain this further, consider the next general example of a probabilistic rule,

$$\mathrm{lrv}_a(n_1, n_2, \dots) \leftarrow_{P(\mathrm{lrv}_a|\mathrm{lrv}_b, \mathrm{lrv}_c, \dots)}, \mathrm{lrv}_b(k_1, k_2, \dots), \mathrm{lrv}_c(m_1, m_2, \dots), \dots, \tag{4}$$

where the left- and right-hand terms are LRVs. The rule declares that the r.v.'s produced by the left-hand LRV having a specific constant as state has a probability that depends on the states of (some of) the r.v.'s produced by that on the left. This probability is placed as a "subscript" to the right of the arrow, which arrow denotes the lifted dependency. Last, it is important to note that the arguments placed in parentheses are logical variables, as in logical functions, that can be instantiated to constant values as dictated by logical rules. The key point here is that an r.v. is created from the left-hand LRV for each possible instantiation of all the logical variables acting as its arguments. The rule also defines which r.v.'s are its parents, i.e., these r.v.'s coming from the right LRVs. Note also that there can be also other logical rules that force the logical variable instantiations to conform to the potential logical rules that contain these logical variables.

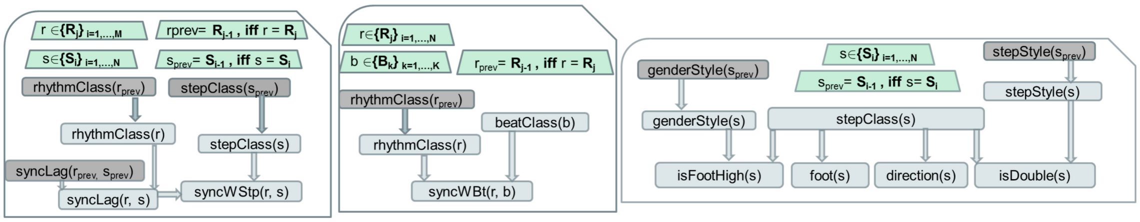


Fig. 1. MEBN model for analyzing Tsamiko dances.

Finally, there are also abstract elements in MEBNs that exist and used in conjunction with these mentioned above:

- MEBN Fragments (MFrags): an MFrag is a container that groups logical functions, LRVs, and probabilistic and logical rules. A MEBN consists of multiple MFrags. The purpose of MFrags is to confine the scope of the logical rules of an MFrag to the logical variables placed as arguments in the LRVs and in the logical rules existing also in this MFrag. Also, MFrags help us to categorize the MEBN elements (LRVs, logical rules, etc.) into groups to be easily understandable and readable.
- Input MFrag nodes and recursion: To declare probabilistic rules across LRVs belonging in different MFrags, one can add in the MFrag in which the child LRV resides the parent LRVs as *input* nodes. Then, the logical rules of this MFrag apply also to the arguments of the input LRV. An input node/LRV in an MFrag can come also from the *very same* MFrag, even being the same LRV. In the latter case, we have the case of *recursion*, which provides us with the flexibility to built r.v. chains coming from the same LRV of arbitrary size. Note that recursion is supported also in DBNs and HMMs. In MEBNs, when used with the aforementioned modeling tools and elements, recursions boost dramatically the modeling flexibility and, as a result, the inference efficiency of MEBNs. MEBN software at the moment supports only first-order recursion. Fortunately, however, this suffices for our modeling needs in the present work.
- MEBN probability definition function: The probability distribution residing at the left arrow of probabilistic rules, such as that in Equation (4), provides the local conditional probability of the regular r.v.'s coming from the left-hand LRV given its parents. In MEBNs, this distribution is *lifted*, in the same sense that the LRVs are lifted r.v.'s, and it is in practice a function that takes as input the parent r.v. states and returns the probability for each potential state of the child r.v. This probability is calculated internally in the probability function using parameters that have predetermined, constant values. In this work, however, we explicitly set the probability, e.g., $P(\mathrm{lrv}|\mathrm{lrv}_b, \mathrm{lrv}_c, \dots)$, to be directly equal to a specific parameter, i.e., $P(\mathrm{lrv}_a|\mathrm{lrv}_b, \mathrm{lrv}_c, \dots) = \theta_{\mathrm{lrv}_a}^{\mathrm{lrv}_b, \mathrm{lrv}_c, \dots}$. Estimating the appropriate values of the probability distribution parameters of all rules is an important contribution of this work, as we shall see in Section 4.

3.2.1 Tsamiko MEBN. In Figure 1, the MEBN model for the Tsamiko dance case is depicted. The green nodes are the logical rules, the small letters denote logical variables, while the big ones denote constants. Also, the light blue are LRVs and the brown-grey are input LRVs. The MEBN depicted in this figure encompasses all the information about Tsamiko conveyed in Tables 2 and 3. More specifically, the explicit information imbued in the model are feature values and relations between events and features that are defined in Tables 2 and 3. Each feature is an LRV while the relations are represented by lifted dependencies that are denoted by directed arrows. Next, we describe the MEBN in details using the FOBL terminology.

Logical FOBL elements:

a. Definition of logical variables: $\mathbf{isA}(s, stepEvent)$, $\mathbf{isA}(s_{prv}, stepEvent)$: s denotes an arbitrary Tsamiko step (event), while s_{prv} is the previous step of s. The stepEvent is the set of constants from which s and s_{prv} takes values, i.e., $s \in stepEvent = \{S1, S2, S3, \dots\}$.

Table 2. Tsamiko Step Cycle (10-step Sequence): Modalities and Events (Entities) and Their Indices

Rhythm		Visual		Audio	
rhythmIndex	rhythmClass	stepIndex	stepClass	beatIndex	beatClass
R1	RA	S1	SA	B1	BA
R2	RB			B2	BB
R3	RB	S2	SB	B3	BC
R4	RA	S3	SC	B4	BA
R5	RB			B5	BB
R6	RB	S4	SD	B6	BC
R7	RA	S5	SE	B7	BA
R8	RB			B8	BB
R9	RB	S6	SF	B9	BC
R10	RA	S7	SG	B10	BA
R11	RB			B11	BB
R12	RC	S8	SH	B12	BC
R13	RA	S9	SI	B13	BA
R14	RB	S10	SJ	B14	BB
R15	RC			B15	BC

Table 3. Class and Style Depended Values of the Tsamiko Step Features

Feature Name	Value	Classes
hasDirection	RightD	SA, SB, SC, SD, SE, SF
	LeftD	SG, SH, SI, SJ
foot	Right	SA, SC, SE, SH, SJ
	Left	SB, SD, SF, SG, SI
isDouble	True	SB, SD, SI and stepStyle is "Double"
	False	otherwise
isHigh	True	SF, SJ and genderStyle is "Female"
	False	otherwise

isA(r, $rhythmEvent$), isA(r_{prv}, $rhythmEvent$): r denotes a rhythm event, while r_{prv} is the rhythm event previous to r. The rhythmEvent is the set of *constants* from which r and r_{prv} takes values, i.e., $r \in \{R1, R2, R3, \ldots\}$.

b. Logical rule: **closeStep(r, s)=True or False**: The rhythm r and step s events are enough close with each other so that their temporal difference must be taken into account (see syncStep below).

<u>*Probabilistic (FOBL) rules and LRVs:*</u>

sclass(s) $\leftarrow_{P(\text{sclass}(s)|\text{sclass}(s_{prv}))}$ **sclass(s_{prv})** : A Tsamiko step (s) class takes 1 of 10 class values, i.e., sclass(s) = SA, SB, …, or SJ, according to the probability $P(\text{sclass}(s)|\text{sclass}(s_{prv}))$ that is conditioned on the previous step (s_{prv}) class sclass(s_{prv}). The exception to this rule is the first Tsamiko step ($s = S1$), whose class is independent of any other step. Note that this is a *recursive* definition.

bclass(b) $\leftarrow_{P(\text{bclass}(b)|\text{bclass}(b_{prv}))}$ **bclass(b_{prv})**: The beat b is one of the BA, BB, and BC classes, i.e., bclass(b) = BA, BB, or BC. The probability $P(\text{bclass}(b)\|\text{bclass}(b_{prv}))$ the beat b to take a certain class depends on the previous beat (b_{prv}) class (bclass(b_{prv})).

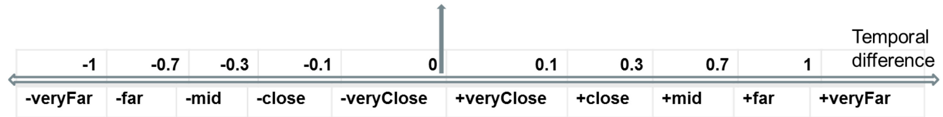


Fig. 2. Range discretization to degrees of closeness (in seconds).

rclass(r) $\leftarrow_{P(\text{rclass}(r)|\text{rclass}(r_{prv}))}$ **rclass(r_{prv})**: The rhythm event r belongs in one of the RA, RB and RC classes, i.e., rclass(r) = RA, RB, or RC. The probability $P(\text{rclass}(r)|\text{rclass}(r_{prv}))$ the rhythm event r to take a certain class depends on the previous (r_{prv}) class (rclass(r_{prv})).

synced(s) $\leftarrow_{P(\text{synced}(s))}$: When the dancer executes a step (s) in synchronization with the rhythm then synced(s) =True, otherwise, the value is False. Note also that there are no dependencies with other LRVs. The synced LRV is a parent of the syncWStp LRV, and, depending on its state, the latter takes values accordingly. Note also that synced does not contain r as an argument, which means that the r.v.'s produced by synced for a specific s will be a parent of all the r.v.'s produced by syncWStp for this specific s and all r that pair with r (see syncWStp's definition).

syncWStp(r, s) $\leftarrow_{P(\text{syncWStp}(r,s)|\text{sclass}(s),\text{rclass}(s),\text{synced}(s))}$ **sclass(s), rclass(r), synced(s)**: syncWStp denotes "synchronization weight for step." This rule applies to a pair of a rhythm event (r) and a Tsamiko step (s), given that closeStep(r, s)=True, i.e., their temporal difference is small enough. More specifically, only pairs for which their temporal difference falls inside a finite continuous and symmetric around zero range of values, are considered close (closeStep(r, s)=True). In our experiments, this range is set to be $[-1, 1]$, and we discretize it to nine values: $\{veryClose, +/- close, +/- mid, +/- far, +/- veryFar\}$, as shown in Figure 2. Then, syncWStp(r, s), which can be seen as the "weight of closeness" between the step and the rhythm event takes one of these values with a probability, which depends on the step class and whether the dancer is synchronized in general with the rhythm. In other words, this LRV models the temporal relation of a step with a rhythm event by characterizing the closeness with each other, and it is conditioned on the event types and whether the dancer manages to be in synchronization with the rhythm (i.e., synced(s) =True).

syncWBt(r, s) $\leftarrow_{P(\text{syncWBt}(r,s)|\text{bclass}(s),\text{rclass}(s))}$ **sclass(s), rclass(r)**: This denotes "synchronization weight for beat." This rule is similar to syncWStp, except that it applies to a pair of a rhythm event (r) and a beat (b). The same discretization with that used for syncWSt is employed.

hasDirection(s) $\leftarrow_{P(\text{hasDirection}(s)|\text{sclass}(s))}$ **sclass(s)**: a Tsamiko step (s) goes either to the right or left direction (hasDirection(s) =RightD or LeftD, respectively), with $P(\text{hasDirection}(s)|\text{sclass}(s))$, which depends on the step class (sclass(s)).

foot(s) $\leftarrow_{P(\text{foot}(s)|\text{sclass}(s))}$ **sclass(s)**: A Tsamiko step (s) is executed either with the right or left foot (foot(s) = RightF or LeftF, respectively), with $P(\text{foot}(s)|\text{sclass}(s))$, which depends on the step class (sclass(s)).

isDouble(s) $\leftarrow_{P(\text{isDouble}(s)|\text{sclass}(s),\text{stepStyle}(s))}$ **sclass(s), stepStyle(s)** : A Tsamiko step (s) is executed either as a double or single step (isDouble(s) = Double or Single, respectively), with probability $P(\text{isDouble}(s)|\text{sclass}(s), \text{stepStyle}(s)))$, which depends on the step class(sclass(s)) and the dance style (stepStyle(s) = Double or Single).

isFootHigh(s) $\leftarrow_{P(\text{isFootHigh}(s)|\text{sclass}(s),\text{genderStyle}(s))}$ **sclass(s), genderStyle(s)** : A Tsamiko step (s) is executed with the foot high with probability $P(\text{isFootHigh}(s)|\text{sclass}(s), \text{genderStyle}(s))$, which depends on the step class(sclass(s)) and the dance style (genderStyle(s)=Male or Female).

stepStyle(s) $\leftarrow_{P(\text{stepStyle}(s)|\text{stepStyle}(s_{prv}))}$ **stepStyle(s_{prv})**: A Tsamiko step s is executed with stepStyle being double or single (stepStyle(s) =Double or Single, respectively), with probability $P(\text{stepStyle}(s)|\text{stepStyle}(s_{prv}))$, which depends on the stepStyle of the previous step (s_{prv}). The stepStyle LRV models that which we call the "style" of the dance regarding the variation of the dance where there is a double step or not. When stepStyle is Double, then the second, fourth, and eighth steps of the cycle are executed as double steps and single

Table 4. Tsamiko Dance Performance Example, Given with FOBL Terminology

Events/features	Extraction method	Description
isA($S1, stepEvent$), ..., isA($SN, stepEvent$)	step detection	$S1, S2, \ldots, SN$ are N step events, ordered by their index.
isA($B1, beatEvent$), ..., isA($B, beatEvent$)	beat detection	$B1, B2, \ldots, BM$ are M step events ordered by their index.
isA($R1, rhythmEvent$), ..., isA($RM, rhythmEvent$)	expert knowledge	R1, R2, ..., RM are M rhythm events ordered by their index.
closeStep($R1, S1$), closeStep($R2, S2$), ..., closeStep(RM, SN)	absolute temporal difference < threshold	Tsamiko step-rhythm events pairs that are close, i.e., with sufficiently small temporal differences.
syncWStp($R1, S1$)=VeryClose, syncWStp($R2, S2$)=VeryClose, ..., syncStep(RM, SN) =VeryClose	discretization of temporal difference	The (discretized) degree of closeness between Tsamiko step and rhythm events that are close.
closeBeat($R1, B1$), closeBeat($R3, B3$), ..., closeBeat(RM, BM)	absolute temporal difference < threshold	beat-rhythm events pairs that are close.
syncWBt($R1, B1$)=VeryClose, ..., syncWBt(RM, SN) =VeryClose	discretization of temporal difference	The (discretized) degree of closeness between beat and rhythm events (see closeBeat input).
isDouble($S1$)=True/False, ..., isDouble(SN)= ...	classification	the isDouble medium-level features, assigned with certain values.
isFootHigh($S1$)=True/False, ..., isFootHigh(SN)=True/False	classification	the isFootHigh medium-level features, assigned with certain values.
foot($S1$)=Right, ..., foot(SN)=RightF/LeftF	classification	the foot medium-level features, assigned with certain values.
hasDirection($S1$)=RightD, ..., hasDirection(SN)=RightD/LeftD	classification	the hasDirection medium-level features, assigned with certain values.

otherwise. Even though there is no formal definition about what is a dance style in general, according to Kaeppler (2001), we use this term to denote the variations of a specific dance. When we want to declare the style of a Tsamiko dance, we give both kinds of style (i.e., stepStyle/genderStyle) separated by a backslash, e.g., male/single.

genderStyle(s) $\leftarrow_{P(\text{genderStyle}(s)|\text{genderStyle}(s_{prv}))}$ **genderStyle(s_{prv})**: A Tsamiko step s is executed with genderStyle being male or female (genderStyle(s) =Male or Female, respectively), with probability P(genderStyle(s)|genderStyle(s_{prv})), which depends on the genderStyle of the previous step (s_{prv}). When genderStyle is true, then the 6th and 10th step of the cycle are executed by having the foot raised (i.e., high) and not raised otherwise.

In Table 4, we provide the general form of the input provided to the MEBN, derived from the pre-processing stage of an arbitrary Tsamiko step sequence of N steps, executed in parallel with an M beat sequence. We use the terminology and notation used also above.

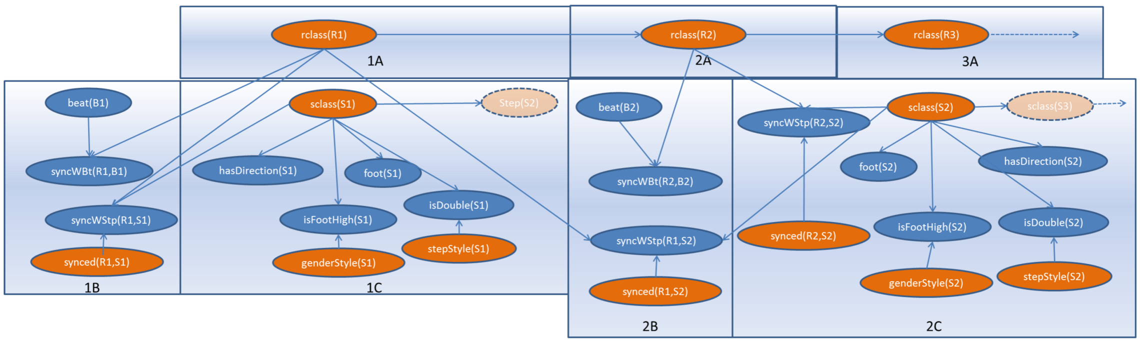


Fig. 3. A part of the SSBN that has been produced by the Tsamiko MEBN model presented in Figure 1. The letters A and B denote the MEBN fragment from which the nodes originate and the numbers denote the ordering of the nodes. Blue nodes are known random variables and orange are considered unknown.

Table 5. Types of Salsa Steps Performed by the Dancers in Our Salsa Dance Dataset

No.	Name	#		No.	Name	#		No.	Name	
C1	forward-basic	143		C2	backward-basic	71		C3	right-turn	1
C4	cross-body	1		C5	preSuzieQ	24		C6	suzie-q-a	28
C7	suzie-q-b	28		C8	preDoubleCross	24		C9	double-cross-a	25
C10	double-cross-b	26		C11	pachanga-tap-a	9		C12	pachanga-tap-b	9
C13	swivel-tap-a	9		C14	swivel-tap-b	7		C15	cross-body-a	9
C16	cross-body-b	9		C17	turn-call	14		C18	girl-turn	14
C19	turn-hand-switch	12		C20	girl-turn	12		C21	cross-body-c	9
C22	cross-body-d	10								

Based on the input described in Table 4, the resulting SSBN has a form that is explained in Figure 3, where the r.v.'s created by the LRVs (depicted as nodes) and the parent–child relationships of the nodes (depicted as arcs). Moreover, the MFrag from which they originate and their temporal ordering are explained.

3.2.2 Salsa MEBN. In the case of Salsa dances, the MEBN model that has been used in our analysis is similar to that of Tsamiko, with the main difference being the step having only one feature, i.e., the class. More specifically, as with Tsamiko, there are three modalities, the visual, the audio, and the rhythm modality. In the visual and audio modalities, the steps of the dancer and the musical beats are manifested as events. Moreover, in this case, there are 22 types of Salsa steps (classes), shown in Table 5.

Moreover, in our experimental dataset, a dancer executes one of three choreographies, each one uniquely predefined by a sequence of 12 steps, where each step is of a specific type. Note that some step types can appear more than once in a choreography, while some others in none.

Figure 4 depicts the MEBN that has been used in the analysis of Salsa dances. More specifically, the utilized MEBN model consists of two MFrags. Moreover, the following list provides information for each MFrag of Figure 4.

<u>*Logical FOBL elements*</u>: the same with the Tsamiko MEBN case.

<u>*Probabilistic (FOBL) rules and LRVs*</u>:

$\mathbf{sclass}(s) \leftarrow_{P(\mathbf{sclass}(s)|\mathbf{sclass}(s_{prv}),\mathbf{choreo}(s))} \mathbf{sclass}(s_{prv})$: A Salsa step ($s$) class takes 1 of 12 class values, i.e., sclass(s) = SA, SB, …,SK, SL according to the probability $P(\text{sclass}(s)\|\text{sclass}(s_{prv}))$ that is conditioned on the

Table 6. 12 Steps in a Salsa Dance in the Three Choreographies and Their Corresponding Type of Executed Steps

Step		Choreography			Rhythm		Beat	
index	type	1	2	3	Event	Class	Event	Class
S1	Step1	C1	C1	C1	R1	RA	B1	BB
S2	Step2	C2	C2	C2	R2	RB	B2	BC
S3	Step3	C1	C1	C15	R3	RC	B3	BA
S4	Step4	C5	C11	C16	R4	RA	B4	BB
S5	Step5	C6	C12	C17	R5	RB	B5	BC
S6	Step6	C7	C2	C18	R6	RC	B6	BA
S7	Step7	C1	C1	C19	R7	RA	B7	BB
S8	Step8	C8	C2	C20	R8	RB	B8	BC
S9	Step9	C9	C13	C21	R9	RC	B9	BA
S10	Step10	C10	C14	C22	R10	RA	B10	BB
S11	Step11	C1	C1	C1	R11	RB	B11	BC
S12	Step12	C2	C2	C2	R12	RC	B12	BA

The classes C1, C2,. . . , C22, indicate the step class, as they are defined in Table 5.

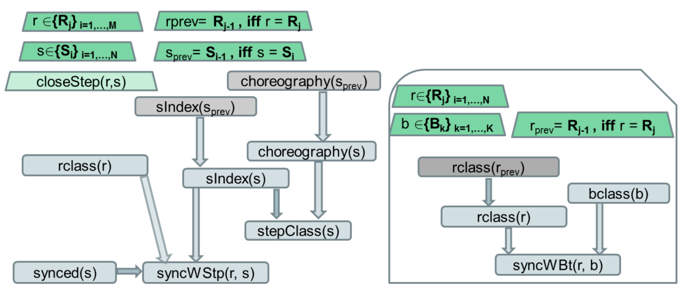


Fig. 4. MEBN model for analyzing Salsa dances.

previous step (s_{prv}) class sclass(s_{prv}) and the choreography (see *choreo* LRV below). The exception to this rule is the first Salsa step ($s = S1$), whose class is independent of any other step. Note that this is a *recursive* definition.

$\mathbf{bclass}(b) \leftarrow_{P(\mathbf{bclass}(b)|\mathbf{bclass}(b_{prv}))}$ $\mathbf{bclass}(b_{prv})$: as with the Tsamiko case, with the difference that there are four classes: BA, BB, BC, and BD.

$\mathbf{rclass}(r) \leftarrow_{P(\mathbf{rclass}(r)|\mathbf{rclass}(r_{prv}))}$ $\mathbf{rclass}(r_{prv})$: as with the Tsamiko case, with the difference that there are four classes: RA, RB, RC, and RD.

$\mathbf{synced}(s) \leftarrow_{P(\mathbf{synced}(s))}$: same with the Tsamiko case.

$\mathbf{syncWStp}(r,s) \leftarrow_{P(\mathbf{syncWStp}(r,s)|\mathbf{sclass}(s),\mathbf{rclass}(s),\mathbf{synced}(s))}$ $\mathbf{sclass}(s)$, $\mathbf{rclass}(r)$, $\mathbf{synced}(s)$: same with the Tsamiko case.

$\mathbf{syncWBt}(r,s) \leftarrow_{P(\mathbf{syncWBt}(r,s)|\mathbf{bclass}(s),\mathbf{rclass}(s))}$ $\mathbf{bclass}(s)$, $\mathbf{rclass}(r)$: same with the Tsamiko case.

$\mathbf{choreo}(s) \leftarrow_{P(\mathbf{choreo}(s)|\mathbf{choreo}(s_{prv}))}$ $\mathbf{choreo}(s_{prv})$: There are three choreographies, i.e., *choreo*$(s) =$ Choreography$< 1, 2, 3 >$. The choreography defines the class of every step of a step sequence.

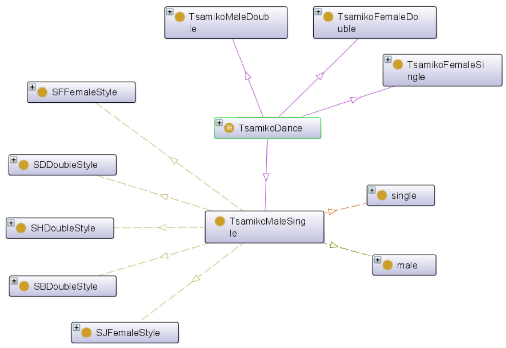
In Table 6, the steps in a salsa dance in the depending on the choreographies are given. In Table 7, we provide the general form of the input provided to the MEBN, derived from the pre-processing stage analysis of an arbitrary Salsa step sequence and the parallel beat sequence. We use the notation that was also used to define the above model.

Table 7. Example of Salsa Dance Performance Pre-processing Results, Given as Input to the MEBN Model

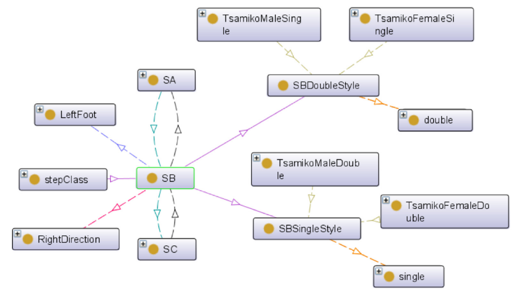
Events/features	**Extraction method**	**Description**
isA($S1, stepEvent$), ..., isA($S12, stepEvent$)	step detection	$S1, S2, \ldots, S12$ are 12 step events order by their index.
isA($B1, beatEvent$), ..., isA($B48, beatEvent$)	beat detection	$B1, B2, \ldots, B48$ are 48 step events order by their index.
isA($R1, rhythmEvent$), ..., isA($R48, rhythmEvent$)	known by expert knowledge	R1, R2, ..., RM are M rhythm events order by their index.
closeStep($R1, S1$), closeStep($R5, S2$), ..., closeStep($R45, S12$)	absolute temporal difference < threshold	Salsa step-rhythm events pairs that are close.
syncStep($R1, S1$)=VeryClose, ..., syncStep($R12, S12$)=VeryClose ...	discretization of temporal difference	The (discretized) degree of closeness between Salsa step and rhythm events (see closeStep input).
closeBeat($R1, B1$), ..., closeBeat($R48, B48$)	absolute temporal difference < threshold	beat-rhythm events pairs with sufficiently small temporal differences.
syncBeat($R1, B1$)=VeryClose, ..., syncBeat($R45, B45$) =VeryClose	discretization of temporal difference	As with syncStep.

3.2.3 System Scalability and Time Complexity. At this point, it would be useful to highlight the advantages and disadvantages of MEBNs. First, MEBNs are based on the production of an SSBN for each situation they are employed. Thus, regarding scalability, the fact that they produce a BN that suits to each specific case enables scale well under certain circumstances that hold much more often than not. More specifically, for the present problem, a MEBN that models a certain type of dance is used as a seed for generating multiple situation-specific models (i.e., SSBNs) for analyzing dance performances executed in many of its variations. Moreover, the MEBN models the fundamental characteristics of the dance with respect to dance steps, music beats, and their synchronization, so it can be used to evaluate a dance performance (in terms of synchronization), independently of its variation, number of steps and accompanying music. In terms of scalability, this is a considerable improvement against straightforward BN approaches that would require the development of a different BN for each case.

Regarding time complexity, the time needed for our MEBN-based approach to yield a result has two parts: (a) the derivation of the SSBN from the MEBN seed and (b) the Belief Propagation algorithm. In the former part, the heavier process relates to the formation of each SSBN node, e.g., their CPTs and links to other nodes. The formation of each CPT depends on the number of states represented by all nodes and the number of connections with other nodes. Thus, for a small number of connections (for example, one to three connections, which is the typical case for the SSBNs generated in our work), we may consider the SSBN creation part to have a close to linear complexity over the number of nodes. With respect to the latter part, the heavier process is the BP algorithm that takes place for estimating the target variables, given the observed ones. We know from the BN literature (Bishop 2006) that if S is the total number of states of nodes in a BN, then the complexity of BP is $O(S^2)$. Note that this complexity does not depend on the number of links between nodes, which is an advantage of this complexity over the number of nodes. However, this comes together with the disadvantage of taking special care to properly design the MEBN so as to avoid an oversized number of connections leading to non-linear complexities when creating the SSBN at the former part mentioned above.



(a) The four possible combination of styles of the Tsamiko dance, i.e., male/single, male/double, female/single, female/double and the relation of the latter with the step classes.



(b) Part of the Tsamiko dance ontology, depicting three steps classes, the subclasses of SB regarding the style, and the relation of these subclasses with the potential styles of the dance

Fig. 5. A part of the Tsamiko ontology.

3.3 MEBNs and Ontologies

At this point, we discuss the connection of MEBNs with ontologies. As also stated in Laskey (2008), a MEBN can be seen as a probabilistic extension of an ontology. This means that there is an ontology that is considered as the deterministic counterpart for each of the MEBNs presented above. We avoid in this work to analyze this thoroughly and provide theoretical justification for this; however, we can note that the functions given in the previous section for both the Tsamiko and Salsa MEBNs are the probabilistic counterparts of the classes and their properties in an ontology (Van Harmelen et al. 2008). Also, the constant values that the logical variables can take are the counterparts ontology's individuals (the logical variables and the functions are instantiated to constants) (Van Harmelen et al. 2008). We also show an ontology for the Tsamiko MEBN and show in Figure 5 a part of the ontology classes (depicted as boxes) and object properties (depicted as arcs). For more information regarding the methodology followed to transcribe information from an ontology to a MEBN see "Probabilistic Ontology: Representation and Modeling Methodology" (Carvalho 2011).

In light of the above, in the general case of using a MEBN instead of an ontology, we can also use MEBNs to encode explicit domain knowledge, as well as merge different MEBNs into a single one or even expand them with new knowledge. Moreover, we can store the data to be analyzed by MEBNs in files with a specified format, which enables us to manage conveniently the data in these files by, e.g., merging, copying, and sharing the files. Moreover, MEBNs come with an application program interface (API) that makes convenient the replacement of the existing ontologies employed in, e.g., ICH preservation platforms and environments.

4 EXPECTATION MAXIMIZATION FOR MEBN PARAMETER LEARNING

4.1 MEBN Parameter Learning Problem Definition

In this section, we derive the Expectation-Maximization algorithm that we employ for learning the MEBN parameters. These parameters, in general, define implicitly the probabilities residing at the probabilistic rules presented in the previous section, see Equation (4).

When it comes to train a MEBN with the EM algorithm, we use example performances as training data so as to adjust the MEBN parameters, where an SSBN for each example performance must be produced. Thus, before we define the problem that we aim to solve with the EM algorithm, it is useful to delve into the details of the SSBN construction. In what follows, we shall be using the FOBL terminology and the MEBN concepts described in the previous section. A MEBN contains lifted random variables (LRV) used for the r.v. creation. More specifically, for an arbitrary SSBN construction case, an LRV plays the role of the "blueprint" used for the creation of multiple, regular r.v.'s. Moreover, the probabilistic rules produce probabilistic dependencies, i.e., the parent–child relations among the r.v.'s. Thus, the above information, as well as the conditional probabilities that are provided for each r.v. by the relevant MEBN functions, suffice to construct the SSBN.

Let L be the number of r.v.'s of all the SSBNs, constructed by example data and used in the context of an arbitrary training case. We denote by $\mathbf{X}^j$ the jth r.v., where $j = 1, \ldots, L$. Also, for $k = 1, \ldots, K$, we denote by $\mathcal{X}^k$ the set of the r.v.'s that originate from the same kth LRV, each r.v. residing either at the same or different SSBN(s). Thus, for a certain k, it is $\mathbf{X}^j \in \mathcal{X}^k$, for all $j \in \mathcal{S}_k$, where $\mathcal{S}_k$ is the set containing all superscripts of the r.v.'s in $\mathcal{X}^k$. As a result, all r.v.'s in $\mathcal{X}^k$ share the same probability values and have the same number and type of parents. Last, all the parents of all r.v.'s in $\mathcal{X}^k$ originate also from the same LRV.

In this work, we assume only discrete r.v.'s, which means that every $\mathbf{X}^j$, for $j = 1, \ldots, L$, is a random variable that takes one out of N_j discrete states. This also means that each r.v. is in essence an SSBN node that has a *local* conditional probability table (CPT). Moreover, we use the following convention to denote the assignment of one of the N_j potential states to an r.v. Each $\mathbf{X}^j$ r.v. is an $N_j \times 1$ vector and, to denote that the r.v. takes the ith state, we set $\mathbf{X}^j(i) = 1$, while all other vector elements are set equal to *zero*. Last, the conditional probability of $\mathbf{X}^j$, which depends on the parent variables $\mathbf{Pa}^j$, is given by the following equation:

$$P\left(\mathbf{X}^j|\mathbf{Pa}^j\right) = \prod_{i=1}^{N_j} \left(\theta_i^{k|\mathbf{Pa}^j}\right)^{x^j(i)}. \tag{5}$$

The right-hand side of the above equation is a categorical distribution, which is a special case of the multinomial distribution (Bishop 2006), where there can be only one of multiple outcomes. Also $\theta_i^{k|\mathbf{Pa}^k}$, for $i = 1, \ldots, N_j$, are the parameters that define the probability that $\mathbf{X}^j(i) = 1$, for every $j \in \mathcal{S}_k$ and for a fixed state configuration of the parents $\mathbf{Pa}^k$. These parameters determine the conditional distribution of every $\mathbf{X}^j = 1$ and in an explicit manner, i.e., each probability $P(\mathbf{X}^j(i)|\mathbf{Pa}^j)$ is equal to the corresponding parameter $\theta_i^{k|\mathbf{Pa}^k}$. Last, denoting by $\bar{\mathbf{X}} = \{\mathbf{X}^j\}_{j=1,\ldots,L}$ the set containing all the random variables of all the SSBNs, their joint distribution, based on Equation (5) and the directed and acyclic properties of BNs, is given by the following product:

$$P\left(\bar{\mathbf{X}}\right) = \prod_{j=1}^{L} P\left(\mathbf{X}^j|\mathbf{Pa}^j\right) = \prod_{j=1}^{L}\prod_{i=1}^{N_j} \left(\theta_i^{k|\mathbf{Pa}^k}\right)^{x^j(i)}. \tag{6}$$

In what follows, for simplicity we write θ_i^k, instead of $\theta_i^{k|\mathbf{Pa}^k}$.

4.2 The Algorithm

In this section, we present the EM algorithm, applied for the MEBN parameter learning task. Following the ML estimation (MLE) paradigm, the parameter estimation problem is formulated as the following (log-)likelihood maximization problem:

$$\hat{\Theta} = \underset{\Theta}{\operatorname{argmax}} \log P(\bar{\mathbf{Z}}; \Theta), \text{ s.t. } \sum_{i=1}^{N_k} \theta_i^k = 1, \; k = 1, \ldots, K, \tag{7}$$

where Θ is the set of all parameters, i.e., $\Theta = \{\theta_i^k\}_{\forall i,k}$.

Note that maximization of a function is equivalent to the maximization of its logarithm. Also, $P(\bar{\mathbf{Z}}; \Theta)$ is the marginal likelihood. To analyze this further, an r.v. can be either an observed or hidden (unknown). Let $\bar{\mathbf{Y}}$ and $\bar{\mathbf{Z}}$ be the set of the hidden and observed random variables of all SSBNs constructed by the example data, respectively.

This means that $\bar{\mathrm{Y}} \subset \bar{\mathrm{X}}$, $\bar{\mathrm{Z}} \subset \bar{\mathrm{X}}$, $\bar{\mathrm{Y}} \cup \bar{\mathrm{Z}} = \bar{\mathrm{X}}$, and $\bar{\mathrm{Y}} \cap \bar{\mathrm{Z}} = \emptyset$. Thus, the data likelihood in Equation (7) is produced by marginalizing $\bar{\mathrm{Y}}$ from the joint distribution (complete data likelihood): $P(\bar{\mathrm{Z}};\Theta) = \sum_{\bar{\mathrm{Y}}} P(\bar{\mathrm{Y}},\bar{\mathrm{Z}};\Theta)$, where the summation runs over all the configurations of $\bar{\mathrm{Y}}$.

We employ EM to solve the maximization (MLE) problem defined in Equation (7). EM, in essence, is an abstract, iterative algorithm, which, when applied to a specific optimization problem, provides concise steps to derive an iterative algorithm that solves MLE problems. Next, we derive the parameter update equations that are used in an iterative fashion, where at each iteration the likelihood increases. Each iteration has two steps, the expectation and the maximization step (E-step and M-step, respectively). At each iteration (t), and specifically at the M-step, the following lower bound of the marginal log-likelihood in Equation (7):

$$\mathcal{L}\left(\Theta,\Theta^{(t-1)}\right) = \mathrm{E}_{[P(\bar{\mathrm{Y}}|\bar{\mathrm{Z}};\Theta^{(t-1)})]}[\log P(\bar{\mathrm{Y}},\bar{\mathrm{Z}};\Theta)] - \sum_{k=1}^{K}\lambda^k \sum_{i=1}^{N_k}\left(\theta_i^k - 1\right), \tag{8}$$

where $P(\bar{\mathrm{Y}}|\bar{\mathrm{Z}})$ is the posterior computed at the (previous of M) E-step, is maximized with respect to the parameters:

$$\left[\Theta^{(t)},\Lambda^{(t)}\right] = \underset{\Theta,\Lambda}{\operatorname{argmax}}\,\mathcal{L}\left(\Theta,\Theta^{(t-1)}\right),\ \text{where } \Lambda = [\lambda^1,\ldots,\lambda^k]^T. \tag{9}$$

$\Theta^{(t)}$ and $\Lambda^{(t)}$ are the parameter and Langrange mutlipliers update equations at step (t), respectively. Note that the posterior of Y given $\bar{\mathrm{Z}}$ plays a key role here. Next, we summarize the EM algorithm applied to the present problem:

- Iteration t=1: Initialize Θ: set $\Theta^{(1)}$ to guessed values
- For $t = 2,\ldots,T_{\max}$ (where $T_{\max}$ is the maximum number of iterations allowed) or until a convergence criterion is met (i.e., further increase of the data likelihood is insignificant), perform the E and M steps:
  Expectation Step (E-Step): Using $\Theta^{(t)}$, calculate the posterior of the data:

$$P\left(\bar{\mathrm{Y}}|\bar{\mathrm{Z}};\Theta^{(t-1)}\right) = \frac{P\left(\bar{\mathrm{Y}},\bar{\mathrm{Z}};\Theta^{(t-1)}\right)}{P\left(\bar{\mathrm{Z}};\Theta^{(t-1)}\right)}. \tag{10}$$

  Maximization Step (M-Step): Find the update of Θ and Λ by solving the maximization problem in Equation (7) in conjunction with the the posterior calculated in the previous step.
- Denoting by T the last iteration, $\Theta^{(T)}$ are taken as the parameter estimates.

Next, we derive the parameter updates $\Theta^{(t)}$ and $\Lambda^{(t)}$ of the above algorithm. The computation of the posterior is performed in the E-step, for which task we choose the well-known Belief Propagation algorithm, and specifically an implementation based on the junction trees concept (Jensen and Jensen 1994). However, we have to note that it is sufficient to calculate with the BP algorithm the values of the marginal probability of each variable conditioned on the observations. Using the definition of the categorical distribution defined in Equation (5), the expectation of the log-likelihood with respect to the posterior is as follows:

$$\sum_{\bar{\mathrm{Y}}} P(\bar{\mathrm{Y}}|\bar{\mathrm{Z}})\log P(\bar{\mathrm{Y}}|\bar{\mathrm{Z}}) = \sum_{k}\sum_{j\in\mathcal{S}_k}\sum_{i=1}^{N_k} P(\mathrm{Y}_i^j,\mathbf{Pa}^j|\bar{\mathrm{Z}})\log\left(\theta_i^k\right)^{\mathrm{y}_i^j} + \sum_{k}\sum_{j\in\mathcal{S}_k}\sum_{i=1}^{N_k}\log\left(\theta_i^k\right)^{\mathrm{z}_i^j}. \tag{11}$$

Note that the index k takes values for which θ_i^k corresponds to a hidden and an observed variable, for the first and second summation, respectively, although not explicitly mentioned at the boundaries of the summation. Then, for the case where k corresponds to a hidden variable, we equate the derivative of Equation (8), calculated

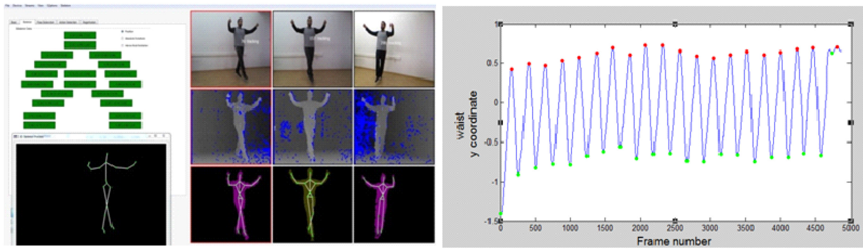


Fig. 6. Tsamiko dance, video Kinect skeleton data, and output of the step event detection algorithm. In the right illustration, the speed of the dancer's waist point is plotted. The maximum and minimum points correspond to Tsamiko steps (events), something that is attributed the nature of the Tsamiko dance.

using Equation (11), with respect to θ_i^k to zero and solve the resulting equation to find the parameter updates:

$$\left(\theta_i^k\right)^{(t)} = \frac{\sum_{j \in \mathcal{S}_k} P\left(\mathbf{Y}_i^j, \mathbf{Pa}^j | \bar{\mathbf{Z}}\right) \mathbf{y}_i^j}{\left(\lambda^k\right)^{(t)}}, \text{ where } \left(\lambda^k\right)^{(t)} = \sum_{i=1}^{N_k} \left(\theta_i^k\right)^{(t)}, \tag{12}$$

ensuring that θ_i^k sum to one for a fixed k.

Last, the update for the case where the parameter θ_i^k corresponds to an observed variable is obtained by taking the derivative of Equation (11) with respect to θ_i^k, setting it equal to zero, and solving the equation w.r.t. the parameter as follows:

$$\left(\theta_i^k\right)^{(t)} = \frac{\sum_{j \in \mathcal{S}_k} \mathbf{z}_i^j}{\left(\lambda^k\right)^{(t)}}, \text{ where } \left(\lambda^k\right)^{(t)} = \sum_{i=1}^{N_k} \sum_{j \in \mathcal{S}_k} \mathbf{z}_i^j. \tag{13}$$

5 EXPERIMENTAL STUDY

5.1 MEBN Training and Inference

Before we proceed to the presentation of the experimental results, we discuss in a consolidated manner the general procedure of applying MEBNs for our problems. More specifically, the MEBN-based analysis process consists of two separate tasks. The first task is the training of the MEBN model with the EM algorithm of Section 4 using a suitable training dataset. The second task is the analysis of a new/unseen dance performance using the trained MEBN and the evidence obtained from the elementary concept detectors. We use the classifier in Kitsikidis et al. (2014) to classify the Tsamiko steps and thus extract their features, see Table 3, and the classification method in Karavarsamis et al. (2016) for the Salsa step classes, see Table 5. For both Tsamiko and Salsa, we use for the detection of beats and their classification to classes the work in Dupont et al. (2009). We have to note at this point that since the features are extracted by Kinect sensors, there can be inaccuracies in the information extracted by these sensors and, thus, erroneous values to these features. However, we deal with this problem efficiently by employing MEBNs, i.e., a probabilistic model, which takes into account the possibility of having such errors, when providing a solution. In practice, in MEBNs, feature values coming from classification, as mentioned above, are modeled as LRVs, which means they are considered to take values randomly, and, thus, they can be erroneous with a probability. This probability is calculated with the EM algorithm. In Figure 6, a Tsamiko expert dancer is depicted in various positions, along with the skeleton data captured while dancing. Also, in Figure 7, the footwork followed by a Salsa dancer when executing the step belonging to the "basic step" class are depicted (Essid et al. 2013).

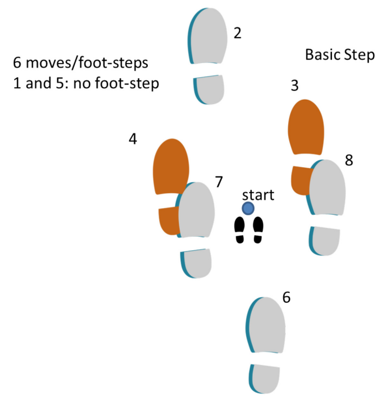


Fig. 7. Steps followed by a Salsa dancer when executing the "basic step".

The MEBN is trained once, using Tsamiko and Salsa dances as training data. Then, once the MEBN model has been trained, it can be used for the analysis of any new dance performance. For the training, we use the algorithm of Section 4, for which we need a training dataset consisting of example dance performances. The reason we apply the EM algorithm is that there are missing values in our data, i.e., the Tsamiko and Salsa step classes. All other values are considered known, i.e., either from expert knowledge or classification, see Tables 4 and 7, or by expert-provided annotation, i.e., the dance style(s), e.g., genderStyle, stepStyle, and choreo and temporal information. This information is taken from ground-truth data found in Kitsikidis et al. (2014), i.e., the time of occurrence of Tsamiko steps, and in Essid et al. (2013), i.e., the starting and ending of each Salsa step. The former source of information, however, is not a product of an expert's manual annotation, but it is extracted using a heuristic, as explained also in Figure 6. However, the ground-truth data we use for Salsa, found in Essid et al. (2013), provide a temporal segmentation of the Salsa performance using the information about the period of the step occurrence provided by an expert. Note also that, when testing a MEBN, the difference with training, is that the style and choreo variables are considered unknown. Last, the result of the training process is the estimation of the MEBN parameter values that best fit to the example performances. Finally, the estimation of these parameters makes the overall process of training and, next, analyzing the new/unseen performances efficient and fully automated.

The training sets we use for the Tsamiko and Salsa consist of recorded dance performances, each one coming either from an expert or an amateur dancer. Particularly for Tsamiko, there are 21 performances, where 8 of them are from experts. The information dictating whether dancer is an expert or an amateur is provided by a Tsamiko expert teacher. For Salsa, however, there are 57 performances but without knowing *a priori* which dancer is an expert or not. To overcome this shortcoming, we followed an automatic procedure that decides which performances belong to experts or to amateurs based on a baseline method.

In our training experiments, to simulate the distinction between the training and testing set, we employ a cross-validation methodology (Geisser 1993), where the main idea is to use a subset of the dataset for training and the rest for testing. Naturally, in our problem, a subset of the expert performances could comprise the training set, while the rest of the expert and amateur performances would comprise the testing set. However, to fully exploit the available data, we employ the leave-one-out cross-validation approach (Bishop 2006), which is also suitably modified so as to conform to the specific nature of our evaluation setting. More specifically, every performance existing in the available dataset, coming either from an expert or amateur dancer, is used for testing. However, in case of an expert performance, it is tested using a MEBN model that has been trained using all the expert performances except for the tested performance. However, in the case of an amateur performance, the model has been trained using all the expert performances. In this way, either we are testing an expert or an amateur performance, we are able to simulate the case where we apply the MEBN-based analysis on a new/unseen performance that has not been used in the training. Moreover, we manage to avoid using amateur performances as training data, something that would lead to erroneously trained models.

5.2 Analysis Objectives

The MEBN models presented in Section 3 have been built to facilitate the following analysis objectives. More specifically, our evaluation scheme has the purpose to test the MEBN models in a twofold manner: first, with respect to their ability to assess the proficiency (expertise) level of the dancer and then his/her synchronization with the music. The immediate results of the MEBN-based analysis of a dance performance are the inferred probabilities that can lead to the recognition of the dance style (Tsamiko) or choreography (Salsa). These are used in a second stage for the assessment of the dancer proficiency level. This assessment is based on the premise that the higher the classification accuracy, the more advanced the dancer is evaluated. Finally, we also assess the synchronization of the dancer with the music. For the two aforementioned evaluations objectives, we employ two metrics that are described and defined in what follows.

Proficiency level assessment (PLA): The objective is to evaluate whether the dance steps in a performance are executed correctly and in the proper order, according to the style/choreography. Note, however, that our ultimate goal is not to classify the entire performance with respect to the style. Instead, our goal is to assess the proficiency level of a dancer, i.e., how skilled (s)he is to execute the dance movements correctly and in proper order. PLA has two stages. In the first, Bayesian Inference applied on the SSBN, which is produced by the MEBN and the test data, to obtain the posterior probability of the dance style given the observations. In the second stage, the PLA metric is calculated using the previously mentioned probabilities, i.e., (a) the *stepStyle* and *genderStyle* for Tsamiko and (b) the *choreo* node for Salsa. In this way, we assess the proficiency level of a dancer not at the level of geometrical properties of the dance performance but in terms of more abstract levels that closer to the human intuition.

More specifically, PLA is calculated by averaging the posterior probabilities of the style nodes. These probabilities indicate the plausibility of the hypothesis that the steps are indeed executed according to the actual style of the dance. More specifically, let us denote with $S = \{S_1, S_2, \ldots, S_N\}$ the sequence of steps constituting the dance performance. Let us also denote with $T = \{T_1, T_2, \ldots, T_K\}$ the set of existing style types. PLA, for the step sequence S, with respect to the step style T_j, is defined as follows:

$$\mathrm{PLA}(S, T_j) = \frac{1}{N} \sum_{i}^{N} P(\mathrm{style}(S_i) = T_j | \bar{\mathbf{Z}}), \tag{14}$$

where $\bar{\mathbf{Z}}$ are the observations (i.e., elementary concepts). More specifically, T_j takes one of the values given in Tables 2 and 6 for Tsamiko and Salsa, respectively. Note that, in Tsamiko, not all styles are mutually exclusive with the others, i.e., a dance has two style types. This means that for a specific Tsamiko performance, there are two PLA scores that we need to calculate to assess the proficiency level, i.e., that $\mathrm{PLA}(S, T_j)$ for which T_j = Male or Female and that $\mathrm{PLA}(S, T_j)$ for which T_j = Double or Single.

Synchronization assessment: The objective is to measure and assess the alignment between the executed dance steps and the beats of the music. In this case, the nodes of interest are those produced by the synced LRV, for both Tsamiko and Salsa, which denotes the degree of simultaneity between the step and the beat events. Let $S = \{S_1, S_2, \ldots, S_N\}$ denote the N step events. In the resulting SSBN, there is one synced(S_j) node for each S_j step. Thus, we define the synchronization assessment score (SAS) as follows:

$$\mathrm{SAS}\,(S) = \frac{1}{N} \sum_{j} (\mathrm{synced}(S_j) = \mathrm{True} | \bar{\mathbf{Z}}), \tag{15}$$

Table 8. Denotations and Styles of the 21 Tsamiko Performances (8 Experts, 13 Amateurs)

Denotation	**Style**		**Denotation**	**Style**		**Denotation**	**Style**
A1	female/single		A8	female/double		E1	female/single
A2	male/single		A9	male/double		E3	male/double
A3	male/single/		A10	male/double		E4	male/double
A4	male/single		A11	male/double		E5	male/double
A5	double/female		A12	male/double		E6	male/double
A6	female/double		A13	male/double		E7	female/single
A7	male/double					E8	male/double

where j runs over the N steps. Higher values for SAS indicate more precise synchronization between the undertaken steps and music beats.

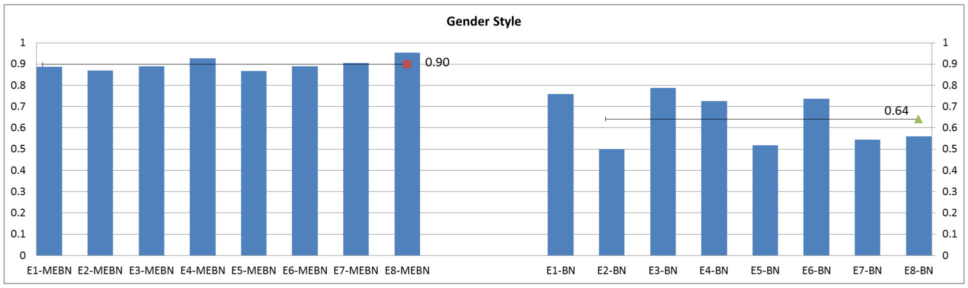
5.3 Tsamiko Dance Analysis Results

In the Tsamiko dances analyzing experiments, we have used 21 recorded performances, all with the same sequence length of 150 steps. Of the 21 recorded performances, 8 were executed by professional dancers (E1–E8), while the rest (A1–A13) were obtained from amateur-level dancers. All performances were executed with the same musical piece, and every performance was annotated with its dance style, as depicted in Table 8.

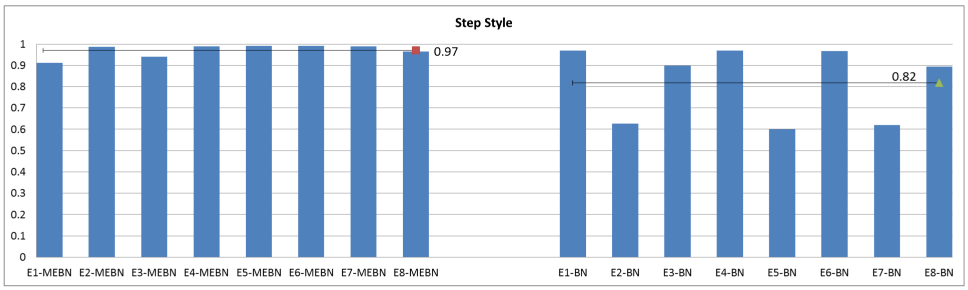
Our purpose in designing the experiments to be conducted for the analysis of Tsamiko dance performances were the following: (a) Using PLA, compare the MEBN-based model of Figure 1 against a standard Bayesian network so as to verify the benefit of MEBNs in adopting to the situation at hand and generating situation specific BNs. (b) Using PLA, compare the performance of an MEBN-based model trained using the learning algorithm of Section 4 against the performance of the same MEBN-based model with its parameters having been set manually. (c) Use the PLA score to distinguish the performances undertaken by amateurs and experts, as presented in Table 8. (d) Verify the ability of our model in assessing the synchronization between the undertaken dances steps and the music beats by comparing the SAS scores for the performances undertaken by amateurs and experts, as presented in Table 8. In this case, as in the previous case, our expectation is that our model will give a high score to the performances undertaken by experts and a lower score to those by apprentices.

5.3.1 MEBNs vs. BNs. To verify the benefit of MEBNs being able to adapt to the situation at hand, we compare our MEBN-based methodology with a straightforward BN. Since the BN-based model cannot adapt to the volatile number of the steps, we are forced to use a BN with fixed size. In the present case, we use a 10-step node BN, which means that the step sequences are being processed by the BN as multiple independent chunks of 10 steps (thus, $N/10$ chunks in total, where N is the step number) instead of as a whole sequence. Also, the comparison takes place only for the expert performances. For a fair comparison, the CPTs of the BN are taken from the trained MEBN network. Figure 8 illustrates the PLA scores obtained in both cases and for all dance performances of Table 8. We can see that the MEBN-based framework provides a consistently higher PLA score, (average: 0.90/0.97, st. dev: 0.03/0.03, for the genderStyle/stepStyle cases, respectively) than the BN approach (average: 0.64/0.82, st. dev: 0.12/0.17). Thus, given that both frameworks incorporate the same information for the observed variables and both frameworks have been designed and trained based on the same domain knowledge, it is reasonable to attribute the consistency of the scoring ability of the MEBN-based framework to handle each performance as a whole and continuous sequence of steps.

5.3.2 EM Training vs. Manually Fixed Parameters. In this experiment, our goal is to verify the efficiency of the EM algorithm presented in Section 4.2 and the usefulness and generalization capability of a trained MEBN



(a)

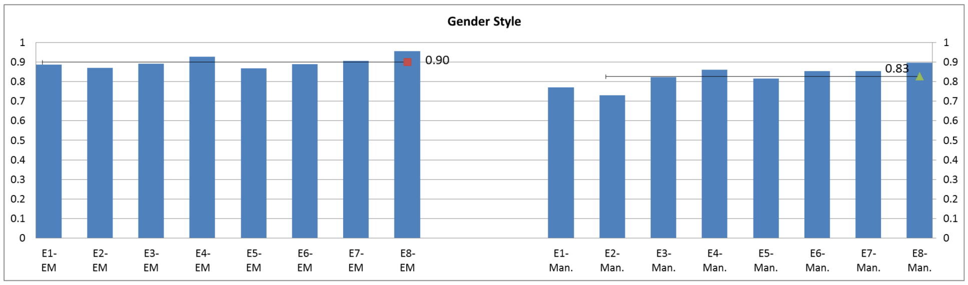


(b)

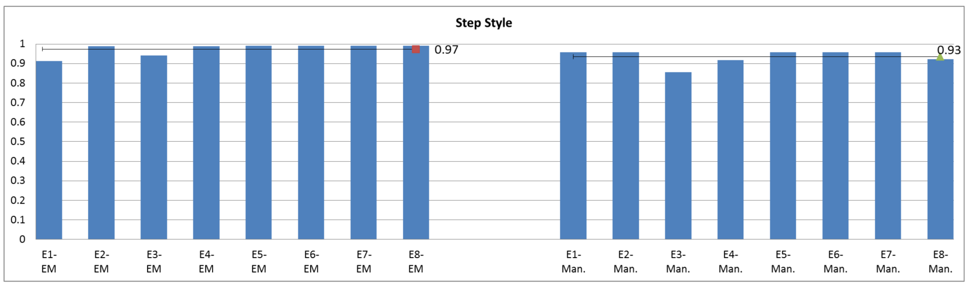
Fig. 8. PLA scores for MEBN vs. BN comparison for two types of style: (a) stepStyle and (b) genderStyle. The average for each case is also given.

model. Figure 9 presents the PLA results achieved by the MEBN model trained using the EM algorithm, against the results of the same MEBN model with its parameters having been set manually to fixed values based on expert knowledge. Also, the comparison takes place only for the expert performances, as above. We can see that the MEBN-based framework provides a more consistent and, in parallel, a higher PLA score than the manual case on average, i.e., EM-trained MEBN average: 0.90/0.97, st. dev: 0.03/0.03, MEBN with manually fixed parameters: average: 0.83/0.93, st. dev: 0.05/0.04, for the genderStyle/stepStyle cases, respectively. Given that both frameworks have the same structure but different parameters, it is reasonable to attribute the PLA accuracy and consistency exhibited by the proposed scoring procedure to capture information that it is difficult to be provided explicitly by an expert.

5.3.3 Experts vs. Amateurs in Proficiency Level Assessment. Our goal in this experiment is to verify the ability of our MEBN-based model in assessing the proficiency level by comparing the PLA score between the performances undertaken by amateurs and experts. Figure 10 presents the PLA scores for the different style combinations by separating between the amateurs and experts. It is evident from this figure that the PLA scores achieved by the experts (marked as E.x) are generally higher than the ones achieved by the amateurs (marked as A.x). Indeed, we can see for the stepStyle the PLA scores achieved by the experts (average score: 0.97, std. dev: 0.03) are evaluated as almost perfect, while many of those achieved by amateurs (average score: 0.88, std. dev: 0.08) are not evaluated with a perfect score or, even worse, are evaluated with a very low score, e.g., A1 and A5. The situation is a bit different in the case of genderStyle, where the expert performances (average score: 0.90, std. dev: 0.03) are assessed as worse, in terms of PLA, than their counterparts in the stepStyle case, something that holds also for the amateurs (average score: 0.83, std. dev: 0.12). This outcome can be attributed to the difficulty of the amateur dancer to perform the male style correctly, which is distinguished from the female style by having the foot high

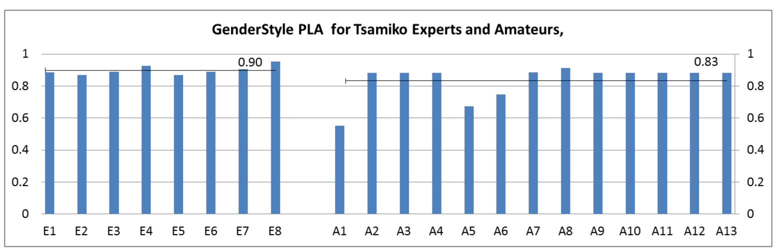


(a)

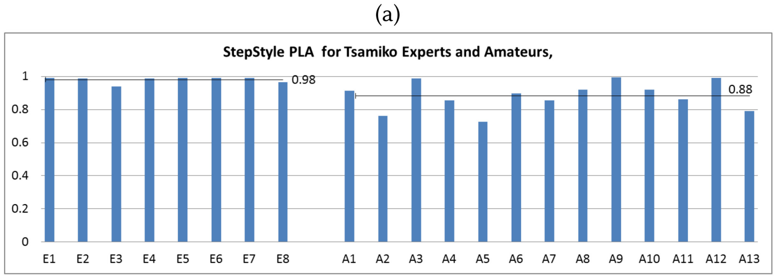


(b)

Fig. 9. PLA scores for trained MEBN vs. fixed valued MEBN comparison for two types of style: (a) stepStyle and (b) genderStyle. The average for each case is also given.



(a)



(b)

Fig. 10. Comparison of PLA scores between amateurs and experts in Tsamiko dance. Average scores are also depicted.

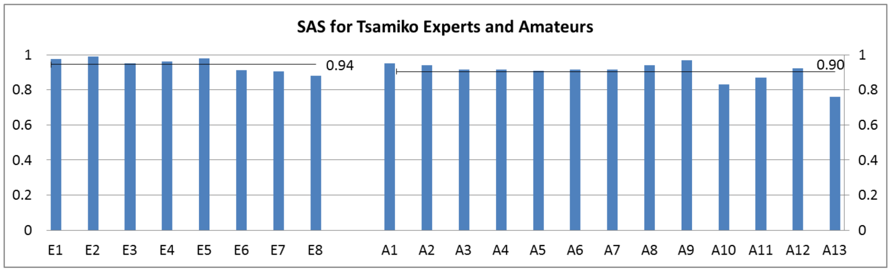


Fig. 11. Synchronization assessment scores for amateurs and experts in Tsamiko dance. Average scores are also depicted.

in some steps. More specifically, the genderStyle nodes define whether the value of the isFootHigh feature should be "true" (for some steps when the style is male) or false (for all steps when the style is female). Given that our analysis methodology managed to distinguish between the amateurs and experts by testing whether this feature had the correct value at each step, it is deduced that amateurs find difficult to have the foot risen properly.

5.3.4 Experts vs. Amateurs in Synchronization Assessment. The last of our experiments performed on Tsamiko dances evaluates the ability of our model in assessing the synchronization between the undertaken steps and music beats. Figure 11 presents the SAS scores for the Tsamiko dance performances of Table 8, distinguishing between amateurs (marked with A.x) and experts (marked with E.x). All expert performances provided high SAS values (with 0.943 average score and 0.039 std. dev), while many amateur performances were assessed with low scores (with 0.904 average score and 0.055 std. dev). The accuracy of the distinction between experts and amateurs based on SAS is evident, though this accuracy is less evident in the case of PLA. However, as we will see next, in the Salsa experiments, this discrimination efficiency is increased in terms of SAS.

5.4 Salsa Dance Analysis Results

For the case of Salsa dance analysis, we adopt the same experimental protocol with Tsamiko and the MEBN model described in Section 3. The 3DLife (Essid et al. 2013) dataset consists of three choreographies, see Table 6, and 57 Salsa dance performances. Each performance consists of 12 Salsa steps, which are executed in the way described in Table 6. Moreover, each dancer executes multiple performances, each one belonging to one of three choreographies. To assess the dance performances, we need, first, to employ the elementary concept detectors to (a) classify each of the 12 Salsa dance steps to one of the step types listed in Table 5; (b) classify each one of the 48 beats, belonging in the audio modality, to one of four types; and (c) identify 48 Rhythm events, which are not calculated based on data analysis but, instead, from the *a priori* known value of the tempo (i.e., the beats per minute rate, bpm), which is 150, 185, and 180 bpm for the first, second, and third choreographies, respectively.

However, in Salsa, we divide the Salsa performances comprising the available dataset, used for training and testing, based on the dancer evaluation method mentioned in Essid et al. (2012). According to this method, evaluation is performed with respect to the dancer's ability to synchronize with the beat and perform correctly the choreography and is calculated from the data. For Salsa, we conducted experiments, presented in what follows, with the following purpose: (a) Verify the ability of our model in correctly recognizing the style followed by each step by comparing the PLA score between the performances undertaken by amateurs and experts, and (b) verify the ability of our model in assessing the synchronization between the undertaken dances steps and the music beats by comparing the SAS scores for the performances undertaken by amateurs and experts.

5.4.1 Experts vs. Amateurs in Proficiency Level Assessment. The objective of this experiment is to assess the proficiency level of the dancer. In this way, we can provide feedback to an amateur when trying to learn the particular Salsa steps and choreographies. Figure 12 presents the PLA scores, obtained as described in Equation (14),

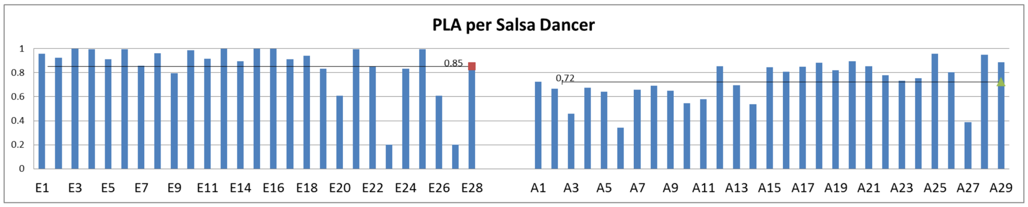


Fig. 12. Proficiency level assessment scores for Salsa dances.

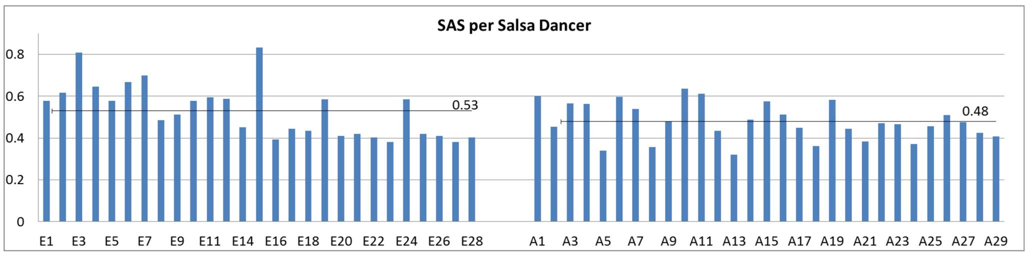


Fig. 13. Synchronization assessment scores for amateurs and experts in Salsa dance.

for $style$ =Choreography$< 1, 2, 3 >$, for all available performances distinguishing between experts (marked as E.x) and amateurs (marked as A.x) dancers. Note that we take the dancers of intermediate level as experts for the training procedure. It is evident that the PLA scores coming from experts (average value is 0.85 and st. dev is 0.21) are, on average, higher than the ones coming amateurs (average value is 0.72 and st. dev is 0.16), advocating the usefulness of our model in performing proficiency level assessment in Salsa dances.

5.4.2 Experts vs. Amateurs in Synchronization Assessment. Figure 13 presents the SAS scores for the analysed Salsa performances, distinguishing between experts (marked as E.x) and amateurs (marked as A.x). It is evident that the SAS scores coming from experts (average score is 0.53, std. dev is 0.12) are, on average, higher than the ones coming from amateurs (average value is 0.48, std. dev is 0.08), advocating the ability of our model to effectively assess the synchronization between steps and music beats.

5.5 Comparing with a State-of-the-Art Method in Dance Synchronization

As a last experiment, we compare the performance of the proposed MEBN-based approach in terms of their synchronization assessment efficiency with a state-of-the art algorithm used in Essid et al. (2012). Both methods are able to produce a ranking of the examined dance performances, which allows their direct comparison with each other based on the gold standard ranking provided by the 3DLife dataset (Essid et al. 2013). Next, we further describe how this comparison is performed.

In the 3DLife dataset, 17 of 57 Salsa dance performances have been scored by experts. Apart from these scores, we also score them with (a) the SAS methodology and (b) the work in Essid et al. (2012), denoted as Virtual dance performance evaluator (VDPE). Note that VDPE is modified for our purposes to take into account only the ability of the dancer to synchronize with the beat. At the same time, we make use of the 17 numerical expert-provided scores, denoted as ES, that are bundled with the 3DLife dataset. Considering the ground-truth scores in 3DLife as a gold standard for establishing a comparison, we use the SAS and VDPE scores so as to evaluate which of these algorithms provides more similar results with the gold-standard ranking.

For the ith performance (of the 17), we denote by SAS_i, SC_i, and ES_i the scores provided by the SAS, VDPE methods and the experts in the 3DLife dataset, respectively. For each of the three scores, namely, SAS_i, SC_i and

Table 9. The Edit and Hamming Distances of SAS and VDPE with Respect to ES Scores

	$A = SAS - s$	$B = VDPE - s$
Edit distance $E(A, B)$	13	16
Hamming distance $H(A, B)$	15	16

ES_i, for $i = 1, \ldots, 17$, we obtain an ordered list of the 17 dance performances, in descending order, and thus obtain three rankings.

Then, we check the similarity of the ranking results. We formally define as similar these rankings that, when provided as input to the two distance metrics described below, produce a high value. The distance metrics are (a) the edit ($ED(A, B)$) and (b) the Hamming ($H(A, B)$) distance (Navarro 2001), defined for the strings A and B. ED is specified as follows. Assuming that each symbol forming the string A and B is drawn from a finite alphabet, the ED from string A to string B is the minimum number of necessary insertions, deletions, and substitutions of symbols belonging in the alphabet that turn A into B. Hamming distance relies on a simpler idea, which is that the distance between the two strings A and B is the count of in-order symbol pairs that differ from each other.

We denote by $SAS - s$, $VDPE - s$ and $ES - s$ the ranked lists, treated as strings for the comparison purposes, corresponding to the SAS, VDPE, and ES, respectively. As shown in Table 9, the ranking of the dance performances produced by the SAS algorithm has an ED equal to 13, whereas the ranking of the performances produced by VDPE has an edit distance of 16. In terms of the Hamming distance, $SAS - s$ and $ES - s$ rankings differ by 1 unit. These results advocate the ability of the proposed method in performing the synchronization assessment of a dance more consistently with the experts view, compared to the algorithm presented in Essid et al. (2012).

6 CONCLUSIONS AND FUTURE WORK

In this article, we have presented and proposed a framework, based on MEBNs, for analyzing manifestations of ICH practices and dance in particular. The efficiency of the analysis was tested on specific dances, i.e., Tsamiko and Salsa. The experimental results demonstrated the accuracy of our dancer assessment methodology, both of his/her ability to execute the moves correctly and be synchronized with the beat of the music. Particularly, we built a MEBN model for each dance case (Tsamiko and Salsa) and use it for the analysis task. We demonstrated how the logical part of MEBNs can be used to encode the expert knowledge about the ICH domains corresponding to the dances. Moreover, we showed that the probabilistic part of MEBNs fosters the capturing of new, unknown, and implicit knowledge, which was impossible to be rigorously provided and defined by experts. We also supported the claim that MEBNs can replace and extend the ubiquitous practice of using ontologies and demonstrated how this can be done. More specifically, we showed that our approach uses MEBNs instead of an ontology to encode domain explicit knowledge.

Lastly, we trained the MEBNs with the EM algorithm, using annotated dance performances as examples. Training entailed the automatic selection of the model parameter values according to the ML criterion. We also derived the EM algorithm used for training. Experiments demonstrated that the PLA and SAS scores, as they are calculated in our semantic analysis using MEBNs trained with the EM algorithm, are accurate and reliable metrics for assessing a dancer, in contrast with the cases where the parameters are adjusted manually (i.e., guessed by an expert) and/or a BN is used instead of a MEBN. Also, in the experiments, it was shown that our methodology manages to distinguish between expert and amateur dances. In addition, the synchronization assessment method was shown to be superior to a state-of-the-art method in terms of the accuracy of the SAS scoring procedure.

Also, we believe that the overarching goal demonstrated in our work, i.e., ICH dances learning, is strongly connected with ICH preservation and the passing of such knowledge to new generations. More specifically, our priority and overarching goal is to use the proposed methodology in an automatic, digital (hardware and software) ICH learning platform dedicated to training ICH practitioners/students. In this platform, the students can receive corrective feedback about their entire performance or about specific parts. Moreover, we plan to use the style information in the more general context of ICH preservation and safeguarding and facilitate convenient access to the recordings via sophisticated indexing.

In the future, we aim to test the dancer assessment methodology on other dance genres or even in singing scenarios. We also plan to employ our methodology to cases not falling into the ICH category. Moreover, it is worth testing the EM algorithm to other cases to demonstrate the importance of the MEBN parameter learning algorithm in even more complex models. Also, a major foreseeable impact of this work is to highlight the benefits of using not only MEBNs but also the general framework of probabilistic ontologies. With this, we aim to fill the serious lack of semantic analysis methods applied to the ICH domain that is based on probabilistic knowledge representation and Bayesian inference.

REFERENCES

Dimitrios S. Alexiadis and Petros Daras. 2014. Quaternionic signal processing techniques for automatic evaluation of dance performances from MoCap data. *IEEE Trans. Multimedia* 16, 5 (2014), 1391–1406.

Hamdi Aloulou, Mounir Mokhtari, Thibaut Tiberghien, Romain Endelin, and Jit Biswas. 2015. Uncertainty handling in semantic reasoning for accurate context understanding. *Knowl.-Based Syst.* 77 (2015), 16–28. DOI: http://dx.doi.org/10.1016/j.knosys.2014.12.025

Andreas Aristidou and Yiorgos Chrysanthou. 2013. Motion indexing of different emotional states using LMA components. In *SIGGRAPH Asia 2013 Technical Briefs*. ACM, 21.

Andreas Aristidou, Efstathios Stavrakis, Panayiotis Charalambous, Yiorgos Chrysanthou, and Stephania Loizidou Himona. 2015. Folk dance evaluation using laban movement analysis. *J. Comput. Cult. Herit.* 8, 4 (2015), 20.

Franz Baader. 2003. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press.

Khaleghi Bahador, Khamis Alaa, O. Karray Fakhreddine, and N. Razavi Saiedeh. 2013. Multisensor data fusion: A review of the state-of-the-art. *Inf. Fusion* 14, 1 (2013), 28–44. http://www.sciencedirect.com/science/article/pii/S1566253511000558

Pierfrancesco Bellini and Paolo Nesi. 2015. Modeling performing arts metadata and relationships in content service for institutions. *Multimedia Syst.* 21, 5 (2015), 427–449.

Christopher M. Bishop. 2006. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg.

Rodrigo de Salvo Braz, Eyal Amir, and Dan Roth. 2008. *Innovations in Bayesian Networks: Theory and Applications*. Springer, Berlin, 289–317.

Rommel Novaes Carvalho. 2011. *Probabilistic Ontology: Representation and Modeling Methodology*. Ph.D. Dissertation. George Mason University.

Jacky C. P. Chan, Howard Leung, Jeff K. T. Tang, and Taku Komura. 2011. A virtual reality dance training system using motion capture technology. *IEEE Trans. Learn. Technol.* 4, 2 (2011), 187–195.

Balakrishnan Chandrasekaran, John R. Josephson, and V. Richard Benjamins. 1999. What are ontologies, and why do we need them? *IEEE Intell. Syst.* 14, 1 (1999), 20–26.

Giannis Chantas, Alexandros Kitsikidis, Spiros Nikolopoulos, Kosmas Dimitropoulos, Stella Douka, Ioannis Kompatsiaris, and Nikos Grammalidis. 2014a. Multi-entity bayesian networks for knowledge-driven analysis of ICH content. In *Proceedings of the European Conference on Computer Vision*. Springer, 355–369.

Giannis Chantas, Spiros Nikolopoulos, and Ioannis Kompatsiaris. 2014b. Multi-entity bayesian networks for treasuring the intangible cultural heritage. In *Proceedings of the 2014 International Conference on Computer Vision Theory and Applications (VISAPP'14)*, Vol. 2. IEEE, 796–802.

Liming Chen and Chris Nugent. 2009. Ontology-based activity recognition in intelligent pervasive environments. *Int. J. Web Inf. Syst.* 5, 4 (2009), 410–430.

Dominique De Beul, Saïd Mahmoudi, Pierre Manneback, and others. 2012. An ontology for video human movement representation based on benesh notation. In *Proceedings of the 2012 International Conference on Multimedia Computing and Systems (ICMCS'12)*. IEEE, 77–82.

A. P. Dempster, N. M. Laird, and D. B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *J. Roy. Stat. Soc. B* 39, 1 (1977), 1–38.

Martin Doerr. 2003. The CIDOC conceptual reference module: An ontological approach to semantic interoperability of metadata. *AI Mag.* 24, 3 (2003), 75.

Angélique Drémeau and Slim Essid. 2013. Probabilistic dance performance alignment by fusion of multimodal features. In *Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'13)*. IEEE, 3642–3646.

Stéphane Dupont, Thomas Dubuisson, Jérôme Urbain, Raphaël Sebbe, Nicolas d'Alessandro, and Christian Frisson. 2009. Audiocycle: Browsing musical loop libraries. In *Proceedings of the 7th International Workshop on Content-Based Multimedia Indexing (CBMI'09)*. IEEE, 73–80.

Katerina El Raheb and Yannis Ioannidis. 2011. A labanotation based ontology for representing dance movement. In *Gesture and Sign Language in Human-Computer Interaction and Embodied Communication*. Springer, 106–117.

Slim Essid, Dimitrios Alexiadis, Robin Tournemenne, Marc Gowing, Philip Kelly, David Monaghan, Petros Daras, Angélique Drémeau, and Noel E. O'Connor. 2012. An advanced virtual dance performance evaluator. In *Proceedings of the 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'12)*. IEEE, 2269–2272.

Slim Essid, Xinyu Lin, Marc Gowing, Georgios Kordelas, Anil Aksay, Philip Kelly, Thomas Fillon, Qianni Zhang, Alfred Dielmann, Vlado Kitanovski, and others. 2013. A multi-modal dance corpus for research into interaction between humans in virtual environments. *J. Multimodal User Interfaces* 7, 1–2 (2013), 157–170.

Slim Essid and Gaël Richard. 2012. Fusion of multimodal information in music content analysis. In *Dagstuhl Follow-Ups*, Vol. 3. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 37–52.

Europeana. 2017. Europeana Data Model Documentation. Retrieved from http://pro.europeana.eu/page/edm-documentation.

Seymour Geisser. 1993. *Predictive Inference*. Vol. 55. CRC Press.

Zoubin Ghahramani. 1998. Learning dynamic bayesian networks. In *Adaptive Processing of Sequences and Data Structures*. Springer, 168–197.

GreekBoston. 2017. How to Dance the Tsamikos (Tsamiko). Retrieved from http://www.greekboston.com/music/dance-tsamikos/.

Karl Grieser, Timothy Baldwin, Fabian Bohnert, and Liz Sonenberg. 2011. Using ontological and document similarity to estimate museum exhibit relatedness. *J. Comput. Cult. Herit.* 3, 3, Article 10 (Feb. 2011), 20 pages. DOI: http://dx.doi.org/10.1145/1921614.1921617

Jun Hu, Yingchun Lv, and Mu Zhang. 2014. The ontology design of intangible cultural heritage based on CIDOC CRM. *Int. J. u-and e-Serv. Science and Technol.* 7, 1 (2014), 261–274.

Charlotte Hug and Cesar Gonzalez-Perez. 2012. Qualitative evaluation of cultural heritage information modeling techniques. *Journal of Comput. Cult. Herit.* 5, 2, Article 8 (Aug. 2012), 20 pages. DOI: http://dx.doi.org/10.1145/2307723.2307727

Daniel Isemann and Khurshid Ahmad. 2014. Ontological access to images of fine art. *J. Comput. Cult. Herit.* 7, 1 (2014), 3.

Finn V. Jensen and Frank Jensen. 1994. Optimal junction trees. In *Proceedings of the 10th International Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann, 360–366.

Adrienne L. Kaeppler. 2001. Dance and the concept of style. *Yearbook for Traditional Music* 33 (2001), 49–63.

Mubbasir Kapadia, I.-kao Chiang, Tiju Thomas, Norman I. Badler, Joseph T. Kider Jr., and others. 2013. Efficient motion retrieval in large motion databases. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*. ACM, 19–28.

Sotiris Karavarsamis, Giannis Chantas, Dimitrios Ververidis, Spiros Nikolopoulos, and Yiannis Kompatsiaris. 2016. Classifying Salsa dance steps from skeleton poses. (2016). In *Proceedings of the14th International Workshop on Content-Based Multimedia Indexing (CBMI'16).*

Suvi Kettula and Eero Hyvönen. 2012. Process-centric cataloguing of intangible cultural heritage. In *Proceedings of the ICOM International Committee for Documentation: Enrich Cultural Heritage (CIDOC'12).*

Alexandros Kitsikidis, Kosmas Dimitropoulos, Stella Douka, and Nikos Grammalidis. 2014. Dance analysis using multiple kinect sensors. In *Proceedings of the 2014 International Conference on Computer Vision Theory and Applications (VISAPP'14)*, Vol. 2. IEEE, 789–795.

Daphne Koller and Avi Pfeffer. 1997. Object-oriented bayesian networks. In *Proceedings of the 13th Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann, 302–313.

Kathryn Blackmond Laskey. 2008. MEBN: A language for first-order bayesian knowledge bases. *Artif. Intell.* 172, 2 (2008), 140–178.

Patrick Le Boeuf. 2012. Towards performing arts information as linked data? In *Proceedings of the SIBMAS 2012 Conference*.

Vincenzo Lombardo, Antonio Pizzo, and Rossana Damiano. 2016. Safeguarding and accessing drama as intangible cultural heritage. *J. Comput. Cult. Herit.* 9, 1 (2016), 5.

Anupama Mallik, Santanu Chaudhury, and Hiranmay Ghosh. 2011. Nrityakosha: Preserving the intangible heritage of indian classical dance. *J. Comput. Cult. Herit.* 4, 3 (2011), 11.

Meinard Müller, Tido Röder, and Michael Clausen. 2005. Efficient content-based retrieval of motion capture data. In *ACM Transactions on Graphics*, Vol. 24. ACM, 677–685.

Gonzalo Navarro. 2001. A guided tour to approximate string matching. *ACM Comput. Surv.* 33, 1 (2001), 31–88.

Luiz Alberto Naveda and Marc Leman. 2008. Representation of samba dance gestures, using a multi-modal analysis approach. In *Proceedings of the International Conference on Enactive Interfaces (Enactive'08)*. Edizione ETS, 68–74.

Cheol Y. Park, Kathryn B. Laskey, Paulo Costa, and Shou Matsumoto. 2013. *Multi-entity Bayesian Networks Learning in Predictive Situation Awareness*. Technical Report. DTIC Document.

Judea Pearl. 2003. Causality: Models, reasoning, and inference. *Econometric Theory* 19, 46 (2003), 675–685.

Avrom J. Pfeffer and Daphne Koller. 2000. *Probabilistic Reasoning for Complex Systems*. Stanford University Stanford.

Lawrence R. Rabiner and Biing-Hwang Juang. 1986. An introduction to hidden markov models. *IEEE ASSP Mag.* 3, 1 (1986), 4–16.

Jörg Rett, Jorge Dias, and J.-M. Ahuactzin. 2010. Bayesian reasoning for laban movement analysis used in human machine interaction. *Int. J. Reason. Based System* 2, 1 (2010), 13–35.

Stuart Russell and Peter Norvig. 1995. *Artificial Intelligence: A Modern Approach*. Prentice-Hall, Egnlewood Cliffs, New Jersey, 27.

Ari Shapiro, Yong Cao, and Petros Faloutsos. 2006. Style components. In *Proceedings of Graphics Interface Conference 2006*. Canadian Information Processing Society, 33–39.

Takaaki Shiratori, Atsushi Nakazawa, and Katsushi Ikeuchi. 2006. Dancing-to-music character animation. In *Computer Graphics Forum*, Vol. 25. Wiley Online Library, 449–458.

John Sowa. 1999. *Knowledge Representation: Logical, Philosophical, and Computational Foundations*. Course Technology.

Steffen Staab and Rudi Studer. 2013a. *Semantic Portals for Cultural Heritage, Handbook on Ontologies*. Springer Science & Business Media.

Steffen Staab and Rudi Studer. 2013b. *Web Ontology Language: OWL, Handbook on Ontologies*. Springer Science & Business Media.

Guoxin Tan, Tinglei Hao, and Zheng Zhong. 2009. A knowledge modeling framework for intangible cultural heritage based on ontology. In *Proceedings of the 2nd International Symposium on Knowledge Acquisition and Modeling (KAM'09)*, Vol. 1. IEEE, 304–307.

Frank Van Harmelen, Vladimir Lifschitz, and Bruce Porter. 2008. *Handbook of Knowledge Representation*. Vol. 1. Elsevier.

Marilena Vecco. 2010. A definition of cultural heritage: From the tangible to the intangible. *J. Cult. Herit.* 11, 3 (2010), 321–324. DOI: http://dx.doi.org/10.1016/j.culher.2010.01.006

Jens Wissmann. 2012. *Chord Sequence Patterns in Owl*. Ph.D. Dissertation. City University London.

Received January 2017; revised July 2017; accepted July 2017