Journal of **Kim Il Sung** University (Natural Science)
Vol. 3 No. 2
Juche103(2014)

Training Scenario Editing based on New Knowledge Structure to Represent Spatio-Temporal Relation between the Objects

Pak Hyon Chol, Ri Ryong Hyon

Abstract We propose the method to model training process performed optionally accompanying the interaction with users and make the training scenario, and propose the estimation method.

Key words training, structural pattern, video database, string

Introduction

The process of training-simulation that requires repetition can be considered not only as Animation in the aspect of computer graphics but also as the flow process of work in the aspect of work flow accompanied an interaction between a man and a machine [1].

Several schemes such as String, Graph and Tree have been proposed for representing the structural pattern and its primitives [1, 2].

Graph [2 — 4] is widely used to represent the technological procedure(or precedence relations) between the tasks by having an edge for each task and its working time.

In this aspect, Graph can represent the flow process of simulation game, but can not represent the primitive action in detail.

String [5 — 8] is to represent the spatial relations between the objects in a symbolic image, many iconic indexing approaches have been proposed, such as 2*D* - *String*, 2*D G-String*, 2*D C-String*, 2*D C+String*, unique-1*D*-based matrix, *GPN* matrix, virtual image, *BP* matrix and 2*D Z-String*.

To represent the spatial and temporal relations between the objects in a symbolic video, many iconic indexing approaches, extended from the notion of 2*D String* to represent the spatial and temporal relations between objects in a video, have been proposed, such as 2*D B-String*, 2*D C-Tree*, 9*DLT* Strings, 3*D-list*, and 3*D C-String*.

A simulation game is different from a symbolic video *string* in the aspect that a simulation game is a primitive action sequence proceeding selectively accompanied with an interaction between a man and a computer.

1. Representing the Process of Simulation Game

Graph can not represent a primitive action in detail, whereas string can represent a primitive action, so we define the process of simulation game as follows.

Definition 1 Let F and G be a set of ranks of vertices and a set of labels of vertices respectively.

An attributed 3*D-Sim-String* is defined by a quadruple: $G = (N, E, g, f)$, where N is a set of vertices, E is a set of edges, g is a mapping from N to G, and f is a mapping from N to F.

As you can see, each vertex as primitive action has both a vertex type and a vertex label.

2. Representing a Primitive Action

In the training-required simulation game, the implementation process of primitive action is a process performing a primitive action i.e. changing the layouts and attributes of the objects based on the observation and recognition.

We define a primitive action as follow on the base of a primitive action and string theory.

Definition 2 Let A^{vk} be a k^{th} object at the bottom of the v^{th} primitive action.

The attributed $Sim-String$ is defined as the sequence of attributed objects with A^{vk} and layout relation R. It is notated as follows.

$$A^{vk} = A_k^{vk} R_k A^{vk} R_{k+1} \cdots A_{m_k}^{vk} \tag{1}$$

where m_k : a number of objects with A^{vk} and hierarchical relation, A_j^{vk} : an attributed object with A^{vk} and hierarchical relation, and $R_1, \cdots, R_{m_{k-1}} \in R(=\{<, =\})$: alocation relation

A rank of string is defined as "(a number of operators '<' before the symbol + 1) + (sum of attribute indices of all symbols)"

Definition 3 The primitive action is defined as a set of attributed *Sim-String* $A^v = \{A^{v1}, A^{v2}, \cdots, A^{vN}\}$ with index v. (N: a number of objects at the bottom)

3. Definition of Similarity for Attributed 3*D-Sim-String*

We define the similarity of attributed 3*D-Sim-String* by edit distance.

For this, we introduce *CS* (*Can Substitute*) to shows the possibility whether a type of vertex can be substituted into another. When the labels of two vertices are different, that is, *CS* can not be defined, the similarity of vertex can not be defined.

Let $G_1 = (N_1, E_1, g_1, f_1)$ and $G_2 = (N_2, E_2, g_2, f_2)$ be the graphs and let $M : N_1 \to N_2$ be a homomorphism mapping the vertices of G_1 to the ones of G_2.

We call $dom(M) = \{n_1 \mid (n_1, n_2) \in M\}$ and $cod(M) = \{n_2 \mid (n_1, n_2) \in M\}$ as a domain of M and a corresponding domain of M respectively.

For $\forall n \in N_1 \bigcup N_2$, if $n \in dom(M)$ *or* $n \in cod(M)$, then the vertex n can be substituted. If a vertex $n_1 \in N_1$ can not be substituted, it is removed from G_1 (or inserted into G_2).

Similarly, if a vertex n_2 can not be substituted, it is removed from G_2 (or inserted into G_1)

As it is similar to the matching of primitive action, we define the similarity of primitive action as follows.

Definition 4 Let $G_1=(N_1,\ E_1,\ g_1,\ f_1)$ and $G_2=(N_2,\ E_2,\ g_2,\ f_2)$ be two attributed *Sim-String*s, Rank (x) be a rank of symbol x, the similarity between $n_1\in N_1$ and $n_2\in N_2$ is defined as follows.

$$sim(n_1,\ n_2)=\begin{cases}\sum_{i=1}^{N}\sum_{j=1}^{M}W_{i,\ j}^{v}\cdot sim_{i,\ j}^{v}(A^{vi},\ B^{vj}): & if\ \ CS(g_1(n_1),\ g_2(n_2))\\ 0 \quad : & \text{the other}\end{cases} \tag{2}$$

where, $W_{i,\ j}^{v}=\dfrac{Rank(A^{vi})}{\sum_{k=1}^{N}Rank(A^{vi})}$: a coefficient of weight,

$sim_{i,\ j}^{v}(A^{vi},\ B^{vj})=1.0-\dfrac{d(Rank(A^{vi}),\ Rank(B^{vj}))}{\max(Rank(A^{vi}),\ Rank(B^{vj}))}$: the similarity between attributed *Sim-String*s A^{vi} and B^{vj}.

If there is none of the mappings $(n_1,\ n_2)\in M$ and $(m_1,\ m_2)\in M$ or there is no edge such as $(n_2,\ m_2)\in E_2$, $(n_1,\ m_1)\in E_1$ of G_1 is removed from G_1 (or inserted into G_2).

The edge removing from G_2 (inserting into G_1) can be defined as above.

A set of such these edges is noted as *iode* (insert or delete edge).

In case that an edge isn't removed or inserted, it is substituted.

As it is corresponding to the matching that evaluates the user's implementation of simulation game, i.e. the grade of skill according to the game scenario, we define the similarity of simulation game as follows.

Definition 5 The similarity of an attributed 3*D-Sim-String* is defined as follows.

$$S(G_1,\ G_2)=W_n\cdot sim(n_1,\ n_2)+W_m\cdot sim(m_1,\ m_2)+W_e\cdot sim(e_1,\ e_2) \tag{3}$$

where $|E_1|$, $|E_2|$: the absolute values of the edges of G_1 and G_2, $S(e_1,\ e_2)=1-\dfrac{|iode|}{(|E_1|+|E_2|)}$: the similarity between e_1 and e_2, W_n,W_m,W_e $(W_n+W_m+W_e=1,\ 0\le W_n,\ W_m,\ W_e\le 1)$: the coefficient for evaluation set by scenario maker.

Conclusion

In the tank-training simulation game, we study our method and compare with the evaluation by teacher.

In the paper, the result of experiment shows that the proposed method has a same efficiency as the subjective evaluation by teacher and that it is more detail in some cases.

References

[1] J. P. Marques et al.; Pattern Recognition(Concepts, Methods and Applications), Springer, 150～158, 2001.

[2] S. K. Chang et al.; In Proceedings of Fall Joint Computer Conference, 79, 2006.

[3] Tao Yang et al.; J. Parallel Distrib. Comput., 64, 921, 2004.

[4] A. L. Rosenberg et al.; IEEE Transactions on Parallel and Distributed Systems, 18, 11, 648, 2007.

[5] S. Y. Alexandrov et al.; Nuclear Physics, 90, 2003.

[6] Ping Yu et al.; Pattern Recognition Letters, 26, 2500, 2005.

[7] Z. Gharibian Masihi et al.; Serbia & Montenegro, Belgrade, 11, 22, 2005.

[8] E. Vellasques et al.; Pattern Recognition, 41, 3044, 2008.