RODRIGO DE DEUS REINALDO

PROPOSTA DE UM MODELO DE DADOS PARA BANCOS MULTÍMIDIA ATRAVÉS DA ATRIBUIÇÃO DE SEMÂNTICA A EVENTOS

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para a obtenção do Título de Mestre em Engenharia.

São Paulo

2005

RODRIGO DE DEUS REINALDO

PROPOSTA DE UM MODELO DE DADOS PARA BANCOS MULTÍMIDIA ATRAVÉS DA ATRIBUIÇÃO DE SEMÂNTICA A EVENTOS

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para a obtenção do Título de Mestre em Engenharia.

Área de Concentração:
Engenharia Mecatrônica.

Orientador:
Prof. Dr. José Reinaldo Silva.

São Paulo
2005

FICHA CATALOGRÁFICA

Reinaldo, Rodrigo de Deus
Proposta de um modelo de dados para bancos multimídia através da atribuição de semântica a eventos / R.D. Reinaldo. – São Paulo, 2005.
p.

Dissertação (Mestrado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos.

1.Vídeo 2.Multimídia 3.Bancos de dados 4.Recuperação da informação I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos II.t.

A todos que direta ou indiretamente
contribuíram na execução deste trabalho.

AGRADECIMENTOS

Ao Prof. Dr. José Reinaldo Silva, pelo apoio e orientação.

Ao amigo Prof. Dr. Marcos Barreto pela atenção e interesse de sua parte na elaboração deste trabalho.

A minha família, em especial a minha mãe, pelo estímulo e incansável apoio.

A Mayra que me acompanhou nos momentos difíceis e nas vitórias ao longo deste estudo.

Aos amigos "loscuervos" com um abraço de gratidão.

RESUMO

O trabalho apresenta e discuti um modelo de dados para bancos multimídia que permite o armazenamento de características de um vídeo possibilitando a extração de trechos de vídeos baseando-se em eventos previamente cadastrados em uma base de conhecimento. Os eventos serão definidos semanticamente e serão identificados nos vídeos a partir da análise dos objetos encontrados nos vídeos e das relações entre eles ao longo do tempo. Um evento pode possuir múltiplas regras o que permite aprimorá-lo com o passar do tempo. Como a arquitetura dos eventos não é fixa existe a possibilidade da sobreposição de eventos. Vídeos de diferentes áreas foram utilizados para demonstrar melhor a característica do modelo. O modelo foi desenvolvido com a intenção de não ser dependente do processamento de imagem podendo ser integrado com diferentes níveis de processamento de imagem. Desta forma, independente de como os trabalhos evoluírem nesta área o modelo poderá ser implementado. O usuário final realiza buscas nos vídeos utilizando diferentes níveis de abstração. A utilização de uma base de conhecimento onde as regras dos eventos são armazenadas é possível reduzir o tempo gasto com indexação dos vídeos.

ABSTRACT

This study presents a video data model for multimedia database that allows the storage of characteristics of a video. The proposed model making possible the extraction of segments of the video basing on the semantics of events registered previously in a knowledge base. The events will be identified considering the analysis of the objects found in the videos and of the relationships among them along the time. An event can has multiple rules and can be improved in the course of time. The architecture of the events is not fixed allowing the overlapping of them. The model was developed with the intention to be independent of the image processing level and can follow its development. Final users access video content and get desired information with different levels of abstraction. The use of a knowledge base where the rules of the events are stored reduce the time spent with indexation and searches in the videos.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

AS – Ação Semântica

AVIS – Advanced Video Information System

BVO – Background Video Object

CAI – Common Appearance Interval

CS – Clipe Semântico

CVTO – Common Video Object Tree

DISIMA – Distributed Image Database Management System

EMU – Elementary Motion Unit

ERU – Elementary Reaction Unit

FST – Frame Segmentation Tree

FVO – Foreground Video Object

GIS – Geographic Information System

GMCO – Greedy Maximum Common Objects

MBR – Minimum Bounding rectangle

MOQL – Multimedia Object Query Language

OVID – Object Oriented Video Information

SQL – Structure Query Language

UML – Unified Modeling Language

1. INTRODUÇÃO

1.1. Motivação

Nos últimos anos observou-se um aumento considerável da capacidade de processamento e armazenamento dos computadores. Com o aumento da tecnologia tornou-se viável o armazenamento de vídeo digital e troca de informações multimídia. Atualmente existem grandes quantidades de vídeos disponíveis nos chamados bancos de dados multimídia que podem ser utilizados nas mais variadas aplicações.

Grandes condomínios e locais públicos possuem sistemas cada vez mais complexos de vigilância e controle de acesso que geram várias horas de gravações de vídeos de segurança.

Os robôs tornam-se cada vez mais robusto e eficientes, e constantemente são escalados para realizar tarefas no lugar do homem. Alguns robôs são guiados por câmeras que indicam a trajetória que devem percorrer. Para dar flexibilidade às ações de um robô é preciso que ele se adapte da melhor maneira ao ambiente, identificando e interpretando o que está ao seu redor.

Na indústria, a utilização de câmeras mudou o conceito de inspeção do processo. Praticamente todos os produtos são verificados no lugar de uma pequena amostra. A extração de informações dos vídeos de inspeção pode ser útil na maximização da eficiência do processo, redução dos custos com perda de peças e re-trabalho.

Na medicina uma série de exames baseados em vídeos está prosperando possibilitando não só o diagnóstico por imagens, mas também o refinamento dos tratamentos já que vários vídeos de diferentes pacientes podem ser comparados.

Existe, portanto uma série de aplicações onde o tratamento de informações de bancos multimídia é necessário. Solicitações baseadas em características de baixo nível (como cor, forma, textura) ficam longe da percepção humana. Os usuários desejam buscas baseadas no conteúdo dos vídeos o que exige uma abordagem diferente das normalmente utilizadas. Neste contexto, um sistema multimídia deve permitir buscas

por eventos, objetos e relacionamentos, ou seja, deve trabalhar com o conteúdo do vídeo.

1.2. Objetivo.

Como nos dias de hoje existem tecnologias para armazenamento de vídeos digitais, é necessário o desenvolvimento de ferramentas para a manipulação e extração das informações contidas nos vídeos. A proposta deste trabalho é apresentar e discutir um modelo que permita o armazenamento não do vídeo, mas de características que permitam a extração de trechos de vídeos que contenham eventos previamente cadastrados em uma base de conhecimento. Os eventos serão definidos semanticamente e serão identificados nos vídeos a partir da análise dos objetos encontrados nos vídeos e das relações entre eles ao longo do tempo. O objetivo do modelo é permitir que as buscas sejam feitas em função de eventos em uma camada intermediária que será o elo entre a base de conhecimento e o vídeo em si.

1.3. Escopo.

Este trabalho assume que os todos objeto de interesse utilizados nos modelos conseguem ser extraído dos vídeos seja automaticamente (através do processamento de imagem) ou manualmente. Neste contexto, objeto de interesse é uma instância de qualquer objeto físico presente no vídeo como, por exemplo, uma pessoa ou um carro. Quanto mais eficiente for o processamento de imagem no reconhecimento dos objetos e de seus atributos, maior será a complexidade das solicitações que o sistema conseguirá atender. O modelo proposto é independente de aplicação e modela apenas o conteúdo semântico de eventos que podem estar presentes nos vídeos ou não. Não faz parte do escopo do trabalho a definição de um tipo específico de banco de dados para o armazenamento das informações. Como pode ser visto nos trabalhos sobre bancos multimídia, o armazenamento das informações pode ser feito em árvores [KÖPRÜLÜ, 04], bancos relacionais e bancos orientados a objetos [OOMOTO, 93]. Da mesma forma, a linguagem que será usada como ferramenta para extração da informação não é definida embora exemplos de

querys sejam utilizados ao longo do texto utilizando as instruções mais comuns de SQL.

Na Figura 1 está desta destacado o escopo do trabalho dentro do contexto de um sistema multimídia.

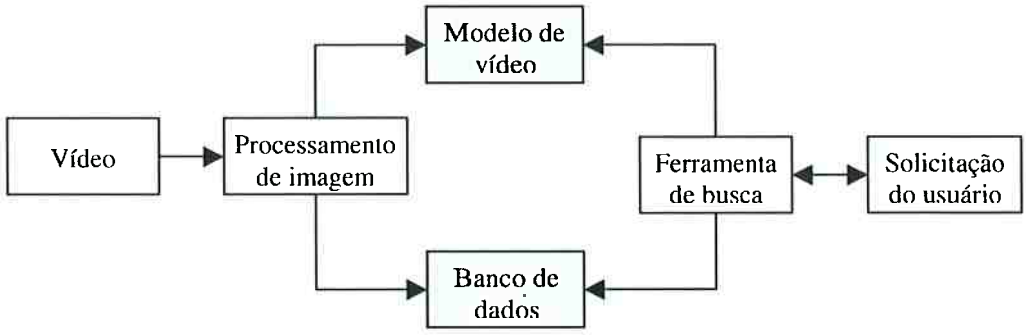


Figura 1 – Definição do Escopo em um Sistema Multimídia.

Um sistema multimídia é composto de uma camada de processamento que identifica as informações contidas nos vídeos. Com estas informações em mãos é possível armazená-las de acordo com um modelo pré-definido. Uma ferramenta de busca também configurada de acordo com o modelo de dados possibilita ao usuário realizar as buscas. O escopo do trabalho cobre a modelagem das informações e como elas podem ser armazenadas em um banco de dados.

O modelo proposto não modela o áudio dos vídeos embora ele possa conter informações que venham a contribuir para a definição semântica de um evento Esta funcionalidade poderá ser adicionada ao modelo no futuro.

2. REVISÃO DA LITERATURA

2.1. Relações entre Objetos de Interesse.

A literatura sobre modelagem de vídeos apresenta como são definidas algumas relações entre os objetos de interesse de um vídeo. Para estes objetos de interesse são atribuídos os seguintes tipos de relacionamento: temporais, espaciais, espaço-temporais e de existência. Nas seções seguintes cada um destes tipos de relacionamento será detalhado.

2.1.1. Relações Temporais.

Pode-se definir um evento com sendo um intervalo onde ocorrem ações semanticamente significativas [EKIN, 04]. Em bancos de dados tradicionais os eventos são indexados pelas datas que eles ocorrem. O problema deste tipo indexação é que nem sempre é possível precisar a data que um determinado evento ocorreu. Este tipo de problema pode ser parcialmente resolvido com uma resposta qualitativa, ou seja, definindo quando um evento ocorreu em relação a outro. Por exemplo: "O evento A ocorreu antes do evento B", "O evento B ocorreu concomitantemente ao evento C".

Nos bancos de dados multimídia é comum um único vídeo possuir uma série de eventos. É preciso então formalizar um relacionamento temporal entre os eventos e entre os objetos que fazem parte dos eventos. Estes tipos de relacionamentos permitem ao usuário realizar solicitações do tipo: "Retorne todos os vídeos que ocorreram antes do evento casamento" ou "Retorne todos os vídeos onde o objeto cadeira aparece antes que o objeto carro".

Em 1983, Allen [ALLEN, 83] definiu uma álgebra para representar as relações temporais entre intervalos de tempo. Sua definição é baseada no fato de que uma condição P que se mantém válida ao longo de um intervalo T, também é válida ao longo de qualquer subintervalo t pertencente a T. Ele definiu 7 relações fundamentais que aparecem descritas na Tabela 1:

Tabela 1 – Relações entre os eventos A e B.

Relação	Símbolo	Símbolo Inverso	Representação
A antes de B (before)	b	bi	AAA BBB
A ao mesmo tempo em que B (equal)	e	e	AAA BBB
A encontra B (meets)	m	mi	AAABBB
A sobrepõe B (overlaps)	o	oi	AAA BBB
A durante B (during)	d	di	AAA BBBBBB
A começa com B (starts)	s	si	AAA BBBBB
A termina com B (finishes)	f	fi	AAA BBBBBB

As relações possuem propriedade transitiva e desta forma a partir destas relações fundamentais é possível inferir relações entre mais de dois intervalos. Por exemplo:

O evento E_r precede o evento E_s:

$$E_r \text{ -- (b)} \rightarrow E_s$$

O evento E_s ocorre durante o evento E_t:

$$E_s \text{ -- (d)} \rightarrow E_t$$

Ou seja:

$$E_r \text{ -- (b)} \rightarrow E_s \text{ -- (d)} \rightarrow E_t$$

A notação utilizada é a mesma utilizada por Allen. O símbolo "--" aparece sempre depois do evento de interesse. O símbolo "→" aponta sempre para o evento utilizado como referência na relação. Com o auxilio da tabela de transitividade [Anexo A] pode-se inferir que, ou o evento E_r ocorreu durante E_t (d), ou ele começou com E_t (s), ou sobrepõe E_t (o), portanto:

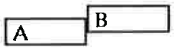
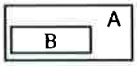
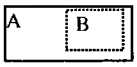
$$E_r \text{ -- (d, s, o)} \rightarrow E_t$$

2.1.2. Relações Espaciais.

As relações espaciais tratam como os objetos se relacionam no espaço. Existem na literatura várias técnicas e nomenclaturas utilizadas para a determinação de relações espaciais entre os objetos. Estas relações podem ser classificadas em relações espaciais quantitativas e qualitativas [LI (A), 96]. As relações quantitativas são definidas a partir da geometria dos objetos, por exemplo, "O objeto A está a 3 metros do objeto B". As relações espaciais qualitativas são calculadas a partir do posicionamento dos objetos, por exemplo, "O objeto A está à direita do objeto B". As relações espaciais também podem ser classificadas como relações topológicas [EGENHOFER, 91] e relações direcionais [LI (A), 96].

Em estudos de sistemas de informação geográfica (GIS), Egenhofer [EGENHOFER, 91] definiu oito relações topológicas que podem ser aplicadas entre dois objetos. As relações definidas por ele se baseiam nas intersecções dos contornos e das áreas de dois objetos. O contorno de um objeto A é a *polyline* constituída da intersecção das fronteiras do objeto A e do complemento do objeto A. A área de um objeto A é a região definida pelo seu contorno. Considerando A^0 e B^0 com sendo as áreas dos objetos A e B respectivamente e δA e δB o contorno de A e B, então a combinação das quatro intersecções ($A^0 \cap B^0$, $A^0 \cap \delta B$ $\delta A \cap B^0$, $\delta A \cap \delta B$) resulta em uma das oito relações topológicas. Dentre as oito relações, duas possuem relação inversa ("cobre" e "coberto por", "contém" e "está contido"). Na Tabela 2 estão representadas as seis relações fundamentais e a combinação das intersecções citadas anteriormente. O símbolo "$\varnothing$" indica uma intersecção vazia e "$\neg \varnothing$" indica uma intersecção não vazia:

Tabela 2 – Relações Espaciais Topológicas definidas por Egenhofer.

Relações	Representação	$\delta A \cap \delta B$	$A^0 \cap B^0$	$\delta A \cap B^0$	$A^0 \cap \delta B$
A e B separados		(∅,	∅,	∅,	∅)
A e B se tocam		(¬∅,	∅,	∅,	∅)
A contém B		(∅,	¬∅,	∅,	¬∅)
A sobrepõe B		(∅,	¬∅,	¬∅,	∅)
A cobre B		(¬∅,	¬∅,	∅,	¬∅)
A é igual a B		(¬∅,	¬∅,	∅,	∅)

Outro processo comum de extrair as relações entre os objetos de um vídeo é através da técnica *Minimum Bounding Rectangle*, MBR. Esta técnica apresenta como vantagem o fato de só precisar de dois pontos para sua representação (os dois pontos que definem o retângulo). Apresenta como desvantagem o fato de não representar com precisão objetos côncavos e objetos na diagonal [LI (A), 96].

Em [PAPADIAS, 94] é utilizada uma técnica onde é detectado um conjunto de pontos especiais chamados pontos representativos e baseado nestes pontos são definidas as relações direcionais. Os pontos representativos podem ser de dois tipos, pontos topológicos e pontos direcionais, e descrevem respectivamente as relações topológicas e direcionais. Considerando-se um MBR que descreve um objeto, o ponto esquerdo inferior e ponto direito superior (que definem o retângulo) são exemplos de pontos representativos direcionais. As relações topológicas são baseadas nas relações definidas em [EGENHOFER, 91].

Em [LI (A), 96] são utilizadas as relações de Allen [ALLEN, 83] para definir relações direcionais entre objetos de interesse em um vídeo. No total são 12 relações de direção:

- Relações estritamente direcionais: norte, sul, leste, oeste.
- Relações compostas: nordeste, noroeste, sudoeste, sudeste.
- Relações de posição: esquerda, direita, em cima, em baixo.

As relações direcionais bem como as 6 relações direcionais definidas em [EGENHOFER, 91] podem ser vistas na Tabela 3 representadas pela álgebra de Allen com o auxílio dos operadores lógicos , $\wedge$ e $\vee$, respectivamente "e" e "ou". A e B são objetos arbitrário e A_x, A_y, B_x e B_y são suas projeções nos eixos *x* e *y*.

Tabela 3 – Relações Espaciais entre dois objetos.

Relação	**Significado**	**Definição com a Álgebra de Allen**
A st B (south)	Sul	A_x {d, di, s, si, f, fi, e} $B_x \wedge A_y$ {b, m} B_y
A nt B (north)	Norte	A_x {d, di, s, si, f, fi, e} $B_x \wedge A_y$ {bi, mi} B_y
A wt B (west)	Oeste	A_x {b, m} $B_x \wedge A_y$ {d, di, s, si, f, fi, e} B_y
A et B (east)	Leste	A_x {bi, mi} $B_x \wedge A_y$ {d, di, s, si, f, fi, e} B_y
A nw B (northwest)	Noroeste	(A_x {b,m} $B_x \wedge A_y$ {bi, mi, oi} B_y) $\vee$ (A_x {o} $B_x \wedge A_y$ {bi,mi} B_y)
A ne B (northeast)	Nordeste	(A_x {bi, mi} $B_x \wedge A_y$ {bi, mi, oi} B_y) $\vee$ (A_x {oi} $B_x \wedge A_y$ { bi,mi }B_y)
A sw B (southwest)	Sudoeste	(A_x {b,m} $B_x \wedge A_y$ {b,m,o} B_y) $\vee$ (A_x {o} $B_x \wedge A_y$ {b,m} B_y)
A se B (southeast)	Sudeste	(A_x {bi,mi} $B_x \wedge A_y$ {b,m,o} B_y) $\vee$ (A_x {oi} $B_x \wedge A_y$ {b,m} B_y)
A lt B (left)	Esquerda	A_x {b, m} B_x
A rt B (right)	Direito	A_x {bi, mi} B_x
A bl B (below)	Em baixo	A_y {b, m} B_y
A ab B (above)	Em cima	A_y { bi, mi } B_y
A ol B (overlap)	Sobrepõe	A_x {d, di, s, si, f, fi, o, oi, e} $B_x \wedge A_y$ {d, di, s, si, f, fi, o, oi, e} B_y

Relação	Significado	Definição com a Álgebra de Allen
A eq B (equal)	Igual	$A_x \{e\} B_x \wedge A_y \{e\} B_y$
A in B (inside)	Dentro	$A_x \{d\} B_x \wedge A_y \{d\} B_y$
A cv B (cover)	Cobre	$(A_x \{di\} B_x \wedge A_y \{fi, si; e\} B_y) \vee (A_x \{e\} B_x \wedge A_y \{di, fi, si\} B_y) \vee (A_x \{fi,si\} B_x \wedge A_y \{di, si, e\} A_y)$
A mt B (meet)	Juntos	$(A_x \{m, mi\} B_x \wedge A_y \{d, di, s, si, f, fi, o, oi, m, mi, e\} B_y) \vee (A_x \{d, di, s, si, f, fi, o, oi, m, mi, e\} B_x \wedge A_y \{m, mi\} B_y)$
A dj B (disjoint)	Separados	$A_x \{b, bi\} B_x \vee A_y \{b, bi\} B_y$

Relações de posição consideram apenas o posicionamento das projeções em um dos eixos. Já as relações direcionais consideram o posicionamento nos dois eixos. É por isso que a representação da relação "esquerda" é diferente da relação "oeste". A relação "esquerda" não considera as projeções no eixo y. e pode coincidir com as relações "oeste", "noroeste" ou "sudoeste".

As relações espaciais encontradas na Tabela 3 serão as utilizadas no modelo proposto por este trabalho.

2.1.3. Relações Espaço-Temporais.

Uma vez definidas as relações espaciais, é possível abstrair um nível a mais e deduzir novas relações levando-se em conta como as relações espaciais entre dois objetos evoluem ao longo de um intervalo de tempo. Se for constatado que a seqüência de relações espaciais entre os objetos O_i e O_j ao longo do intervalo de tempo t foi: "O_i e O_j separados", "O_i e O_j se tocando", "Oi dentro de O_j", pode-se afirmar que durante o intervalo de tempo t, o objeto O_i "entrou" em O_j [CHEN, 2004]. Da mesma forma a análise da distância entre dois objetos ao longo de um intervalo pode determinar se eles estão se aproximando ou se afastando. Um conjunto de relações espaço-temporais pode ser encontrado na Tabela 4 onde as relações estão descritas em função da relações espaciais definidas na Tabela 3. São estas as relações espaço-temporais que serão utilizadas no modelo proposto.

Tabela 4 – Relações Espaço-Temporais baseadas nas relação espaciais.

Relação	Descrição Algébrica
Tocar (Touch)	dj → mt → dj
Pegar (Snap)	dj → mt
Soltar (Release)	mt → dj
Desviar (Bypass)	dj → mt → Release
Entrar (Enter)	dj → mt → ol → $cv^{\leftarrow}$ → in
Sair (Leave)	$Enter^{\leftarrow}$
Cruza (Cross)	Enter → Leave
Fundir (Melt)	dj → mt → ol → eq
Separar (Separate)	$Melt^{\leftarrow}$
Roçar (Graze)	dj → mt → ol → $(cv^{\leftarrow}$ → ol)* → mt →dj

O símbolo '→' indica que há uma seqüência lógica entre as relações espaciais. Uma determinada relação espaço-temporal ocorre se e somente se as relações espaciais ocorrerem na ordem como foram definidas. O símbolo '$^{\leftarrow}$' indica uma relação inversa e o símbolo '*' indica que as relações se repetem [ERWIG, 99].

Sendo assim, pode-se dizer que objetos se "Tocam" quando a seqüência objetos "Separado"(dj) → objetos "Juntos"(mt) → objetos "Separados"(dj) é satisfeita.

Outro tipo de relação espaço-temporal é a que determina a existência ou não dos objetos nos clipes. Em [CHEN, 2004] define-se as relações de existência de objetos em quadros e clipes. A relação "existe" indica quando um objeto pode ser encontrado nos clipes e a relação "não existe" quando um objeto não pode ser encontrado nos clipes. São estas relações de existência que serão utilizadas no modelo proposto pelo trabalho.

2.2. Modelagem de Vídeo

Por um certo tempo, os trabalhos com bancos de dados multimídia podiam ser divididos em dois tipos. Os que faziam o processamento das imagens trabalhando com características de baixo nível como cor e forma e os que trabalhavam com indexação textual das informações [OOMOTO, 93], [DÖNDERLER, 2003]. Quando se trabalha com características como forma, cor e detecção de borda o usuário precisa possuir algum conhecimento de processamento de imagens. Além disso, geralmente um usuário quer saber quais os vídeos em que uma certa pessoa "abre" uma porta e não os vídeos com histograma de cores igual ao de um exemplo. Já os sistemas com indexação textual necessitam de um usuário para realizar a identificação das informações. É possível definir objetos e atributos. As limitações ficam por conta da subjetividade do usuário que registra as características, a dependência do tipo de vídeo e o grande tempo gasto para completar a indexação.

Uma linha alternativa de trabalho propõe sistemas multimídia que envolvem conceito de orientação a objeto e entidade-relacionamento que permitem o gerenciamento da informação contida nos vídeos através da análise do seu conteúdo de forma automática ou manual. Estas técnicas em geral definem entidades e uma hierarquia entre elas para o tratamento da informação. Também são definidas relações entre as entidades que permitem que sejam atendidas requisições, como por exemplo: "Retorne todos os vídeos que contém a pessoa A dirigindo um carro".

Esta última abordagem será a explorada pelo modelo como será visto na seção 3.

2.2.1. OVID.

Em [OOMOTO, 93] é apresentado o modelo *Object-oriented Video Information Database*, OVID. O modelo introduz o conceito de *video object*. Um *video object* é cada seqüência de vídeo com algum significado em especial que é tratada como uma entidade à parte, um objeto, com seus atributos e valores que descrevem seu conteúdo. O sistema não possui uma estrutura hierárquica definida. Toda informação identificada no OVID é indexada textualmente pelo usuário. À

medida que vai assistindo ao vídeo, o usuário identifica e cria os atributos. A generalização dos atributos é que vai determinar a hierarquia entre eles. Um exemplo de generalização pode ser visto na Figura 2 que descreve o vídeo de um documentário sobre estadistas do Japão.

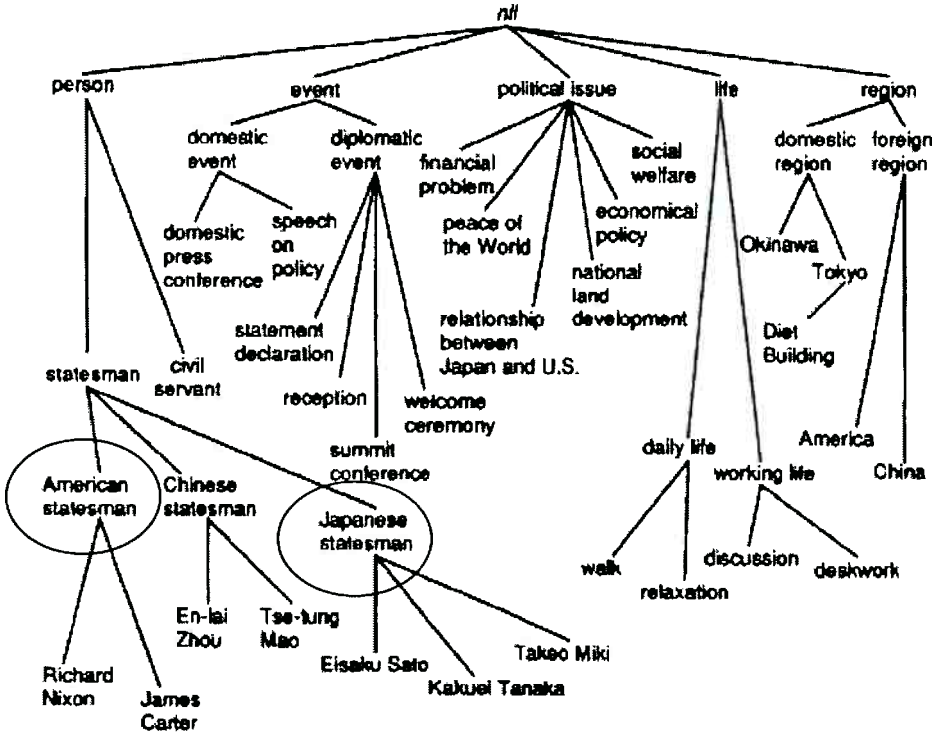


Figura 2 – Generalização de atributos [OOMOTO, 93].

De acordo com a Figura 2 *Japanese stateman* é a generalização dos atributos Kakuel Tanaka, Eisaku Sato e Takeo Miki. Da mesma forma que o atributo *American stateman* é a generalização dos atributos Richard Nixon e James Carter.

Cada *video object* possui um identificador único (o_{id}), um intervalo com uma seqüência de quadros (I) e uma coleção de atributos e seus valores (v) que descreve os quadros. O valor de um atributo pode referenciar outros objetos. Um *video object* é representado por $o_i = (o_{id}, I, v)$. A Figura. 3 mostra a segmentação de um documentário sobre estadistas do Japão em *video objects*.

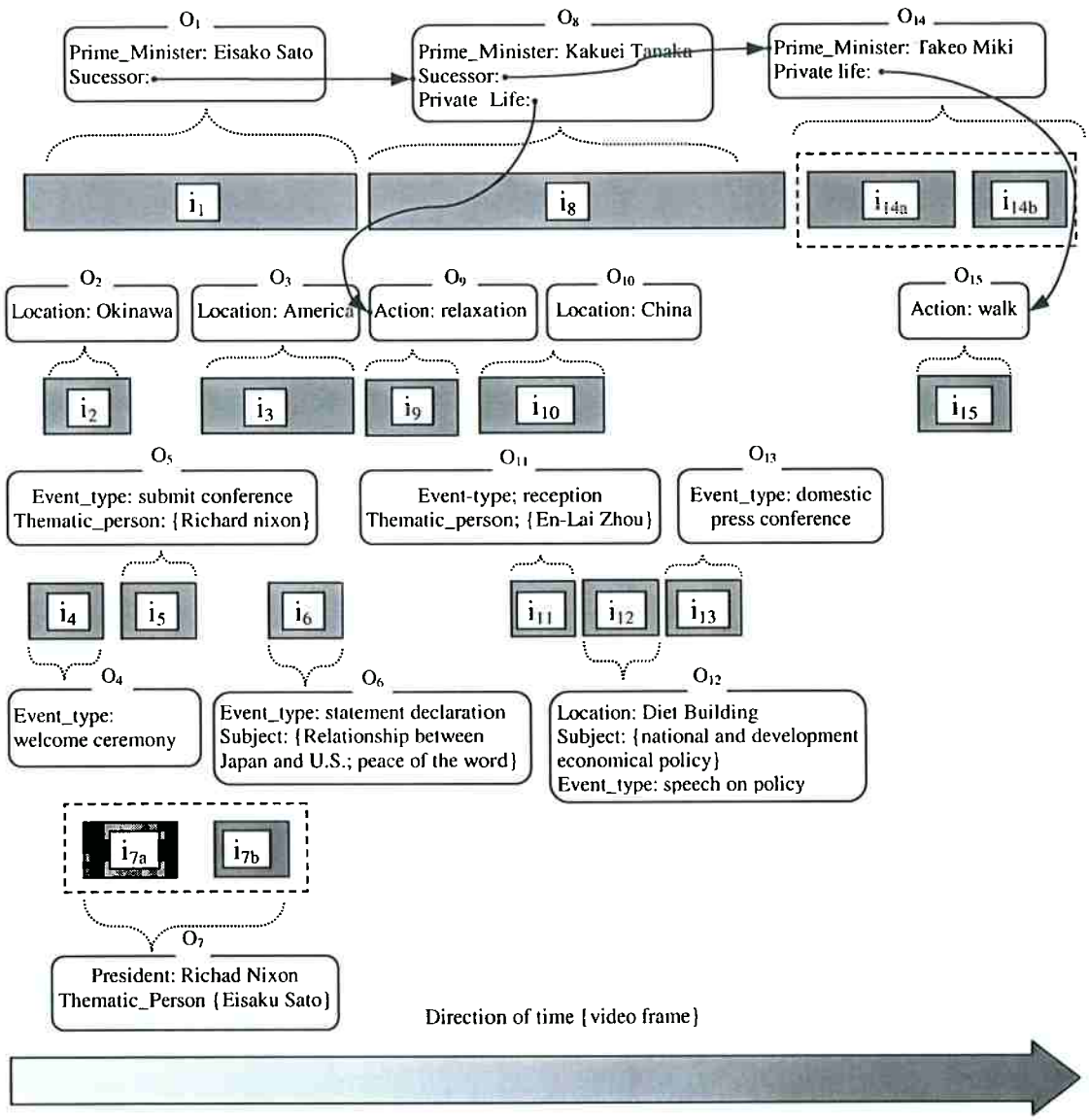


Figura. 3 – Exemplo de segmentação de um vídeo.

Na Figura. 3, $o_8 = (d_8\{i_8\},v_8)$ onde v_8 = [Prime_Minister: Kakulei Tanaka, Sucessor: o_{14}, Private_Life: o_9], $o_9 = (d_9,\{i_9\},v_9)$ onde v_9 = [Action: Relaxation], $o_{14} = (d_{14}, I_{14}\{i_{14a},i_{14b}\},v_{14})$ onde v_{14} = [Prime_Minister: Takeo Miki, Private_Life: o_{15}] e $o_{15} = (d_{15},\{i_{15}\},v_{15})$ sendo v_{15} = [Action : Walk].

O OVID trás operações que permitem a composição de novos objetos a partir de objetos existentes. A operação *Interval Projection* cria um objeto a partir de um subconjunto de atributos de dois objetos. A operação *merge* cria um objeto novo

a partir de dois objetos que possuem dados em comum. A operação *overlap* cria um novo objeto a partir da união dos atributos de dois objetos.

A implementação do OVID é composta de três componentes:

- VideoChart: aplicação que é a interface gráfica que os usuários utilizam para manipular os objetos.
- VideoSQL: aplicação para realização de buscas *ad hoc* para facilitar o retorno das informações.
- Video Object Definition Tool: Aplicação para auxiliar o usuário na definição de objetos.

A linguagem utilizada pelo OVID para realizar as buscas é o VideoSQL. A linguagem é compostas das seguintes estruturas:

- SELECT: especifica somente o tipo de objeto que será retornado (Continuous: somente objetos que estão contidos em uma seqüência, Incontinuous: retorna objetos contidos em mais de uma seqüência de vídeo, anyObject: retorna todos os objetos).
- FROM: especifica o nome do vídeo que se quer analisar.
- WHERE: especifica a condição baseada nos atributos dos objetos.

Um exemplo da busca feita via VideoSQL pode ser visto na Figura 4. No exemplo, são selecionados todos os objetos do vídeo "Prime Minister DB" onde o atributo "Prime-Minister" é igual a "Kakuei Tanaka".

```
SELECT anyObject
FROM Prime Minister DB
WHERE Prime-Minister is Kakuei
Tanaka
```

Figura 4 – Exemplo de busca com o VideoSQL.

2.2.2. CVOT.

Em [LI, 97] e [LI (B), 96] é proposto o modelo chamado *Common Video Object Tree* (CVOT). Este modelo define uma estrutura hierárquica para os vídeos e incorpora os relacionamentos espaciais e temporais entre os objetos do vídeo. O modelo foi desenvolvido visando minimizar dois problemas comuns em outras

modelagens: segmentação de vídeo restritiva, ou seja, segmentação fixa e independente do conteúdo, e o fraco suporte a buscas.

No modelo CVTO, um vídeo é divido em clipes. Um clipe é uma seqüência consecutiva de quadros. A principal característica do modelo é identificar e agrupar clipes que possuam objetos em comum. Este tipo de abordagem permite tratar inclusive sobreposição de clipes.

Na Figura 5 é possível ver uma seqüência de vídeo subdividida em 5 clipes, $C=\{C_1,C_2,C_3,C_4,C_5\}$.

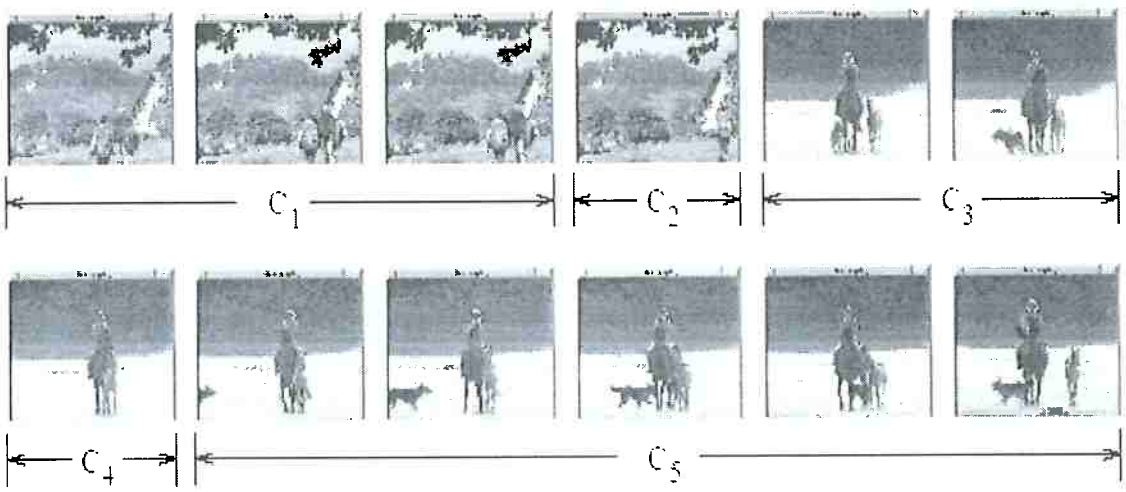


Figura 5 – Segmentação de um Vídeo em Clipes[LI (B), 96].

Dado o conjunto C, define-se o conjunto de objetos de interesse O={João, Maria, casa, cavalo, árvore, cachorro, potro}. Cada clipe possui a seguinte relação de objetos de interesse:

C1 → João, Maria, casa e árvore.

C2 → João, casa e árvore.

C3 → Maria, cavalo, potro e cachorro.

C4 → Maria, cavalo e potro.

C5 → Maria, cavalo, potro e cachorro.

Uma árvore é usada para representar o agrupamento dos clipes como pode ser visto na Figura 6.

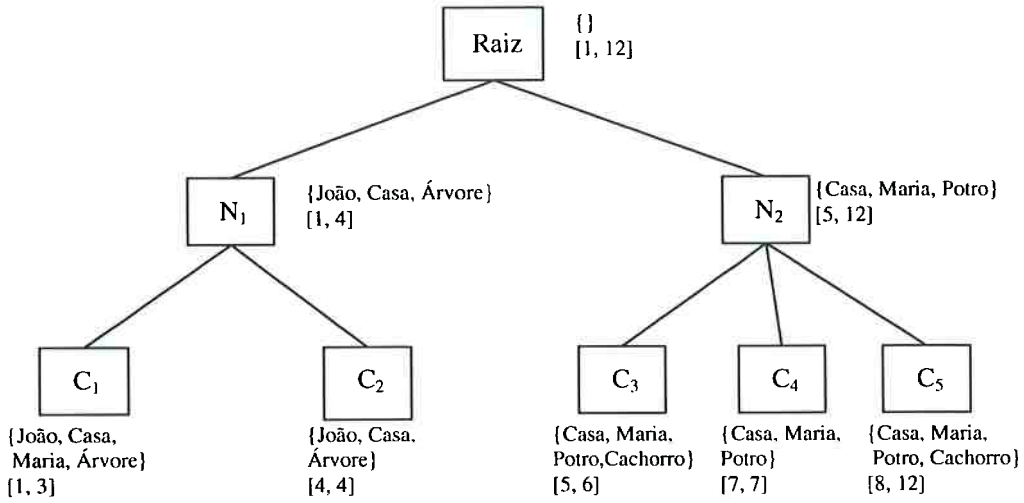


Figura 6 – Árvore Gerada a partir do vídeo da Figura 5.

A árvore é gerada automaticamente por um algoritmo chamado *Greedy Maximum Common Objects* (GMCO). Um nó pai é criado para representar o intervalo de quadros dos nós filhos com referência aos objetos em comum a todos os nós filhos. Desta forma, o nó N_1 na Figura 6 representa o intervalos de quadros dos nós filhos, [1, 4], e possui referência aos objetos em comum entre eles, {João, Casa, Árvore}.

O CVTO é integrado a um modelo de dados, o TIGUKAT que é um banco de dados que incorpora o conceito de orientação a objetos. Ele trabalha com tipos, classes e comportamentos para relacionar os objetos. Por exemplo, "C_pessoa" é uma classe, "B_idade" um comportamento da classe "C_pessoa". Se "joão" for uma instância do tipo pessoa, "joão.B_idade" retorna o valor da idade associado ao objeto "joão".

O autor se baseia nas relações definidas por Allen [ALLEN, 83] para descrever as relações espaciais entre os objetos. São consideradas as 12 relações direcionais e as seis relações topológicas definidas em [EGENHOFER, 91].

2.2.3. CAI.

Em [CHEN, 02] é apresentado o modelo CAI que é um modelo baseado nos *Common Appearance Intervals*, ou seja, são intervalos de quadros onde objetos do interesse aparecem simultaneamente. Por esse modelo um vídeo $Video_i$ é

composto de uma seqüência de clipes $Clip_i$. Neste caso, um clipe é uma abstração de uma tomada ou uma cena. Um clipe $Clip_i$ é composto de uma série de CAIs. Esta seqüência de CAIs observa o aparecimento e desaparecimento dos objetos no vídeo. O modelo apresenta como grande diferença em relação aos outros modelos citados, o fato de modelar separadamente dois tipos de *Video Objetc* (VO), os objetos de fundo, *background video objects* (BVO), e os objetos da cena, *foreground video objects* (FVO). Com isso o modelo é mais eficiente, pois não precisa armazenar propriedades redundantes dos objetos estáticos e suas as relações entre todos os objetos de interesse do vídeo.

O modelo completo é descrito na Figura 7. Um CAI é composto de quadros, pelo conjunto de BVO, FVO e das relações espaciais entres eles (ST). Os FVO são ainda divididos em *Static Video Object* (SVO) e *Motion Video Object* (MVO). Os SVO são compostos pelos MBRs que descrevem os objetos. Os MVO são composto pelos MBRs que descrevem o objeto e pelo *Motion Vector* (MV) que é um vetor que contém a velocidade, a direção e o intervalo do movimento.

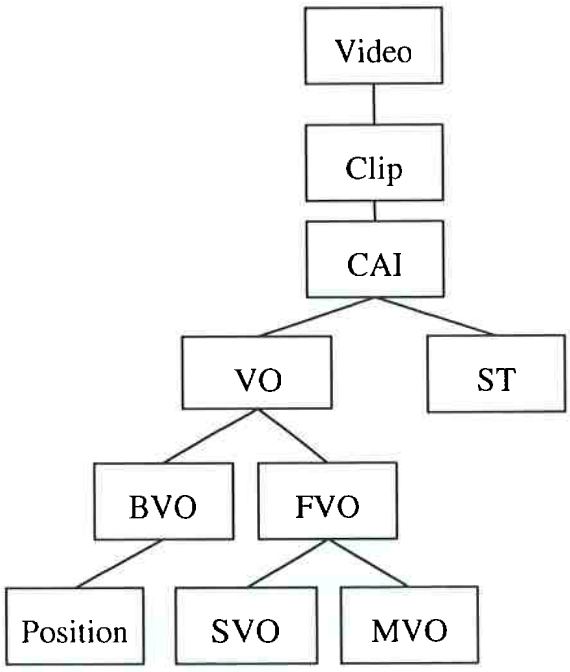


Figura 7 – Modelo Hierárquico do CAI.

As relações entre os objetos de interesse são divididas em dois tipos. As relações entre os BVO e FVO são definidas automaticamente pela propriedade

position dos BVO (*behind,* quando os BVO esta atrás de um FVO ou *full* quando um BVO ocupa toda a cena). As relações espaciais entre os FOV são baseadas nas relações definidas na Tabela 3. As relações são divididas em 12 relações direcionais (norte, sul, leste, oeste, nordeste, noroeste, sudeste, sudoeste, esquerda, direita, em cima e embaixo.) e 8 relações topológicas ("igual", "dentro", "contém", "cobre", "coberto por", "sobrepõe", "juntos" e "separados") As relações espaciais entre dois FVO A_i e A_j em uma série de intervalos de tempo ordenados são chamadas de *st–list* e são descritas por:

$$\text{st–list}_{ij} = [(\alpha_1,\beta_1,I_1),(\alpha_2,\beta_2,I_2),.....,(\alpha_n,\beta_n,I_n)]$$

onde α é uma das 8 relações topológica e β é uma das 12 relações direcionais durante o intervalo de tempo I.

2.2.4. DISIMA Estendido.

Em [CHEN, 04] é apresentado um modelo que é uma extensão do DISIMA, *Distributed Image Database Management System.* O DISIMA originalmente é um modelo que captura a semântica das imagens através dos objetos de interesse, suas formas e suas relações espaciais e temporais com outros objetos de interesse. Esta extensão do modelo consegue atender tanto requisições semânticas ("Selecione todos os vídeos que o objeto O_i aparece") como requisições por características ("Selecione todos os vídeos que contém o objeto O_i com a cor do exemplo dado"). Pelo modelo, um vídeo é dividido em cenas, as cenas são divididas em tomadas e finalmente as tomadas são divididas em quadros. Um quadro específico chamado "quadro chave" é então selecionado para representar as tomadas. Histórias ou cenas são construídas baseando-se no quadro chave. A estrutura hierárquica atribuída ao vídeo é encontrada na Figura 8:

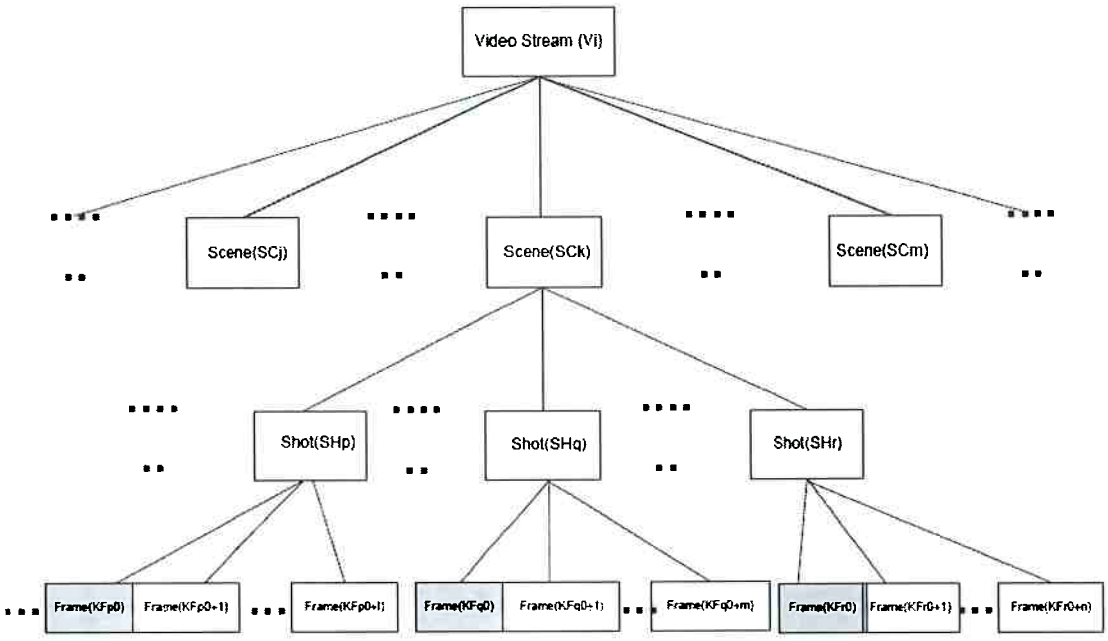


Figura 8 – DISIMA - Estrutura hierárquica de um vídeo[CHEN, 04].

O quadro chave é selecionado para representar o intervalo de uma tomada onde os objetos de interesse nela contidos conservam suas relações espaciais. O quadro chave é selecionado através da observação do aparecimento e desaparecimento dos objetos de interesse nos quadros e das alterações das suas relações espaciais, combinando técnicas manuais e automáticas.

O modelo é composto de dois tipos de objetos de interesse, os objetos físicos e os objetos lógicos. Os objetos físicos são representados pela localização espacial no quadro chave. Os objetos lógicos são utilizados para atribuir semântica ao objeto físico como, por exemplo, nome, dimensão, função.

O modelo apresentado define uma série de predicados para descrever as características dos objetos em um vídeo. Os predicados para descrever as funções de busca são descritos na Tabela 5.

Tabela 5 – Predicados para busca.

Relações de busca	FrameInShot	Retorna os quadros para uma dada tomada.
	ShotInScene	Retorna as tomadas de uma determinada cena.
	SceneInVideo	Retorna as cenas de um determinado vídeo.
	ObjecstInFrame	Retorna o conjunto de objeto contidos em um quadro.
	ObjecstInShot	Retorna o conjunto de objetos contidos em uma tomada.
	ObjecstInScene	Retorna o conjunto de objetos contidos em uma cena.
	ObjecstInVideo	Retorna o conjunto de objetos contidos em um vídeo.
	TrajectoryInFrame	Retorna a trajetória de movimento um objeto em um quadro.
	TrajectoryInScene	Retorna a trajetória de movimento um objeto em uma cena.
	TrajectoryInVdeo	Retorna a trajetória de movimento um objeto em um vídeo.

Para verificar a existência de objeto O_i em um *keyframe* foi criado o predicado keyframe_contains. É possível estender a relação para tomadas e cenas. Um objeto está em uma tomada se estiver presente em todos os quadro chave que fazem parte da tomada. Os predicados que descrevem as relações de existência estão definidos na Tabela 6.

Tabela 6 – Predicados para verificar existência de quadros chaves.

Relações de Existência	keyframe_contains	Retorna verdadeiro se um objeto O_i está contido em um quadro.
	shot_contains	Retorna verdadeiro se um objeto O_i está contido em uma tomada.
	scene_contains	Retorna verdadeiro se um objeto O_i está contido em uma cena.
	video_contains	Retorna verdadeiro se um objeto Oi está contido em um vídeo.

Também foram definidos predicados para descrever o relacionamento espacial entre os objetos. As relações espaciais entre os objeto são divididas em relações direcionais e topológicas. As relações são as mesmas encontradas em [LI (A), 96]. Os predicados que descrevem as relações espaciais são encontrados na Tabela 7. Estas mesmas relações podem ser aplicadas para relacionar objetos de interesse em uma tomada. Como uma tomada é composta de vários quadros, a função deve ser verdadeira para os quadros que fazem parte da tomada.

Tabela 7 – Predicados para as Relações Espaciais entre os objetos.

Relações Direcionais	keyframe_south	Retorna verdadeiro se um objeto Oi está ao sul de um objeto Oj
	keyframe_north	Retorna verdadeiro se um objeto Oi está ao norte de um objeto Oj
	keyframe_west	Retorna verdadeiro se um objeto Oi está a oeste de um objeto Oj
	keyframe_east	Retorna verdadeiro se um objeto Oi está a leste de um objeto Oj
	keyframe_northwest	Retorna verdadeiro se um objeto Oi está noroeste de um objeto Oj
	keyframe_northeast	Retorna verdadeiro se um objeto Oi está nordeste de um objeto Oj
	keyframe_southwest	Retorna verdadeiro se um objeto Oi está sudoeste de um objeto Oj
	keyframe_southeast	Retorna verdadeiro se um objeto Oi está sudeste de um objeto Oj
Relações Topológicas	keyframe_inside	Retorna verdadeiro se um objeto Oi está dentro de um objeto Oj
	keyframe_covers	Retorna verdadeiro se um objeto Oi cobre um objeto Oj
	keyframe_touch	Retorna verdadeiro se um objeto Oi toca um objeto Oj
	keyframe_overlap	Retorna verdadeiro se um objeto Oi está sobreposto pelo objeto Oj
	keyframe_disjoint	Retorna verdadeiro se um objeto Oi está separado de um objeto Oj
	keyframe_equal	Retorna verdadeiro se um objeto Oi é igual ao objeto Oj

Também existem predicados para descrever as relações temporais entre dois objeto de interesse. A relações temporais são baseadas nas 7 relações básicas definidas em [ALLEN, 83]. Os predicados estão definidos na Tabela 8. Os mesmo predicados podem ser aplicados em cenas e vídeos.

Tabela 8 – Predicados para as Relações Temporais entre os Objetos.

Relações Temporais	shot_before	Retorna verdadeiro se a exibição do objeto Oi aparece antes do objeto Oj
	shot_meet	Retorna verdadeiro se a exibição do objeto Oi termina quando começa a exibição do objeto Oj.
	shot_overlap	Retorna verdadeiro se a exibição do objeto Oi termina durante a exibição do objeto Oj.
	shot_during	Retorna verdadeiro se a exibição do objeto Oi começa depois e termina antes que a exibição do objeto Oj
	shot_starts	Retorna verdadeiro se as exibições dos objetos Oi e Oj começam ao mesmo tempo.
	shot_finishes	Retorna verdadeiro se as exibições dos objetos Oi e Oj terminam ao mesmo tempo.
	shot_equal	Retorna verdadeiro se as exibições dos objetos Oi e Oj começam e terminam ao mesmo tempo.

Se as relações temporais entre dois objetos forem analisadas ao longo de um intervalo de tempo, uma seqüência de relações pode ser interpretada como uma ação ou relação espaço-temporal. Por exemplo, a ação "entrar" é composta das relações: objetos separados, objetos juntos, um objeto dentro do outro. A Tabela 9 trás os predicados para algumas ações.

Tabela 9 – Predicados que descrevem Relações Espaço-Temporais.

Relações	shot_enter	Retorna verdadeiro se o objeto Oi entra em Oj.
Espaço/temporais	shot_cross	Retorna verdadeiro se o objeto Oi cruza Oj.
	shot_leave	Retorna verdadeiro se o objeto Oi sai de Oj.
	shot_bypass	Retorna verdadeiro se o objeto Oi dá a volta por Oj.

Todos predicados listados nas tabelas anteriores são utilizados para ampliar as funcionalidades da MOQL (*Multimedia Object Query Language*) que é uma linguagem de busca voltada para banco de dados multimídia utilizada pelo modelo. A Figura 9 trás exemplos de utilização do linguagem MOQL.

- Busca A

Retorne todos as tomadas que contem o ator A

```
SELECT C
FROM Shots C, Actor A
WHERE C contains A
```

- Busca B

Retorne todas as tomadas que contem o ator A entrando no prédio B

```
SELECT C
FROM Shots C, Actor A, Building B
WHERE C contains A
AND C contains B
AND A enters B
```

Figura 9 – Implementação dos predicados shot_contains e shot_enter.

2.2.5. ST-AVIS.

O modelo AVIS [KÖPRÜLÜ, 04], *Advanced Video Information System*, é um modelo que utiliza conceitos de orientação a objetos. O modelo indexa um vídeo baseado nos objetos de interesse, atividades e eventos que são as entidades do

modelo. Os objetos de interesse podem ser um personagem de um filme ou um objeto qualquer (carro, mesa, cadeira). Uma atividade é qualquer ação atribuída a uma seqüência de quadros (andar, correr, comer). Um evento é uma instância de um tipo de atividade. É composto de um tipo de atividade, papéis desempenhados nas atividades, objetos e os atores que desempenham os papéis. Sendo assim, no evento "O atleta está correndo a maratona", a atividade é "correr", o papel é "corredor" e o ator é "atleta". O modelo de segmentação de vídeo proposto trabalha com estas entidades e cria um mapa de associação para identificar em quais quadros os objetos aparecem e os eventos ocorrem. Na Figura 10 temos um exemplo do mapa de associação de um vídeo.

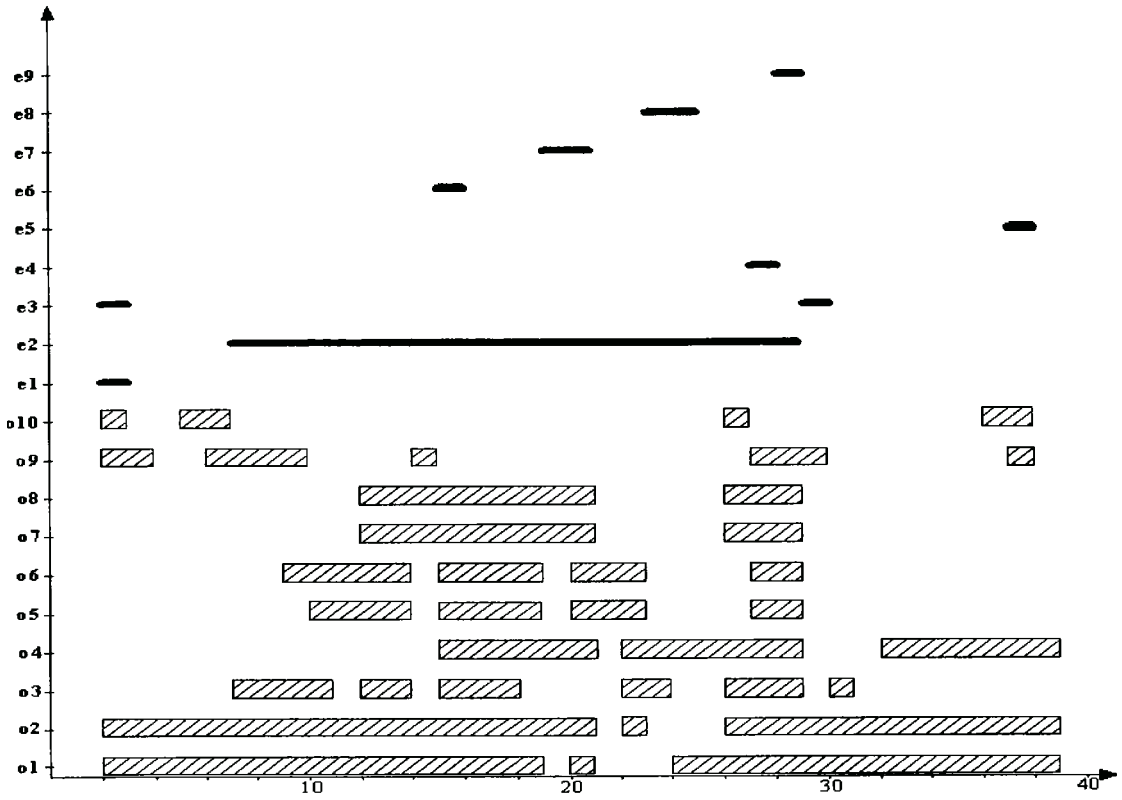


Figura 10 –Exemplo do mapa de associação de um vídeo [KÖPRÜLÜ, 04].

No eixo x estão representados os quadros em ordem cronológica, e no eixo y, estão listadas as entidades que compõe um vídeo. Neste caso só estão representados os objetos e eventos presentes no vídeo. Cada entidade representada possui uma linha que indica em quais quadros ela está presente. Por exemplo, o objeto o_1 está presente nos intervalos de quadros [2, 19], [20, 21] e [24, 39]. O evento e_1 está presente no intervalo de quadro [2, 3].

O mapa é transformado em uma estrutura denominada *Frame Segmentation Tree* (FST) que armazena em cada nó as entidades que fazem parte do intervalo definido pelo nó. A Figura 11 trás a árvore correspondente aos 11 primeiros quadros.

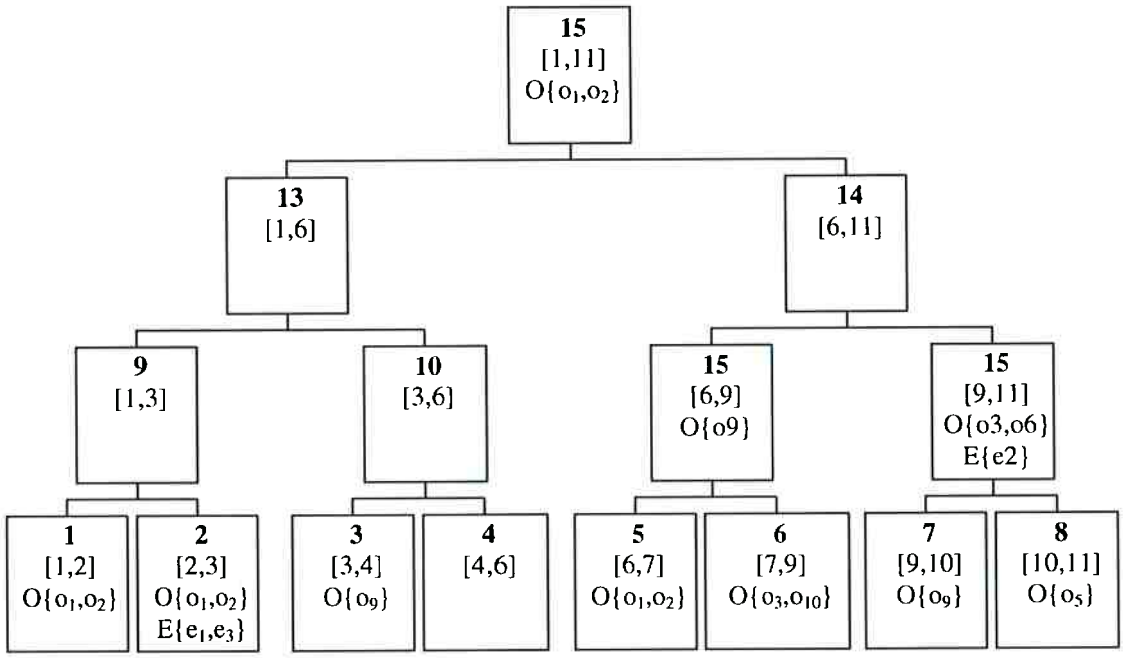


Figura 11 – FST do vídeo no intervalo entre os quadros [1,11].

O nó 2, por exemplo, representa o intervalo de quadros [2,3] e contém o conjunto de objeto O{ o_9, o_{10} } e o conjunto de eventos E{ e_1,e_3 }.

O preenchimento dos nós respeita duas regras:

- Se um objeto excede o intervalo definido pelo nó, então este objeto é representado em mais de um nó.
- Se um objeto não aparece em todos os quadros do intervalo definido pelo nó, então o objeto deve ser representado pelo nós-filhos deste nó.

Embora o AVIS possua grandes capacidades de atender requisições complexas baseadas em conteúdo, ele não é capaz de tratar requisições que envolvam relacionamentos espaciais e espaço-temporais.

O modelo ST-AVIS, *Spatial Temporal Advanced Video Information System*, é uma extensão do modelo AVIS para tratar as relações espaciais entre os

objetos. Ao longo de um intervalo I, um objeto pode ser encontrado em várias regiões. O modelo AVIS tradicional representa todas essas informações em um único nó. No modelo ST-AVIS, um evento é subdividido dependendo da trajetória percorrida pelo objeto, como pode ser visto na Figura 12. Cada nó então contém uma lista de pares {objeto, área}. Cada par contém um objeto e a área ocupada no intervalo definido pelo nó. Como um evento é subdividido e o número de nós aumenta existe um aumento no custo na busca proporcional a log(n) onde n é o número de nós.

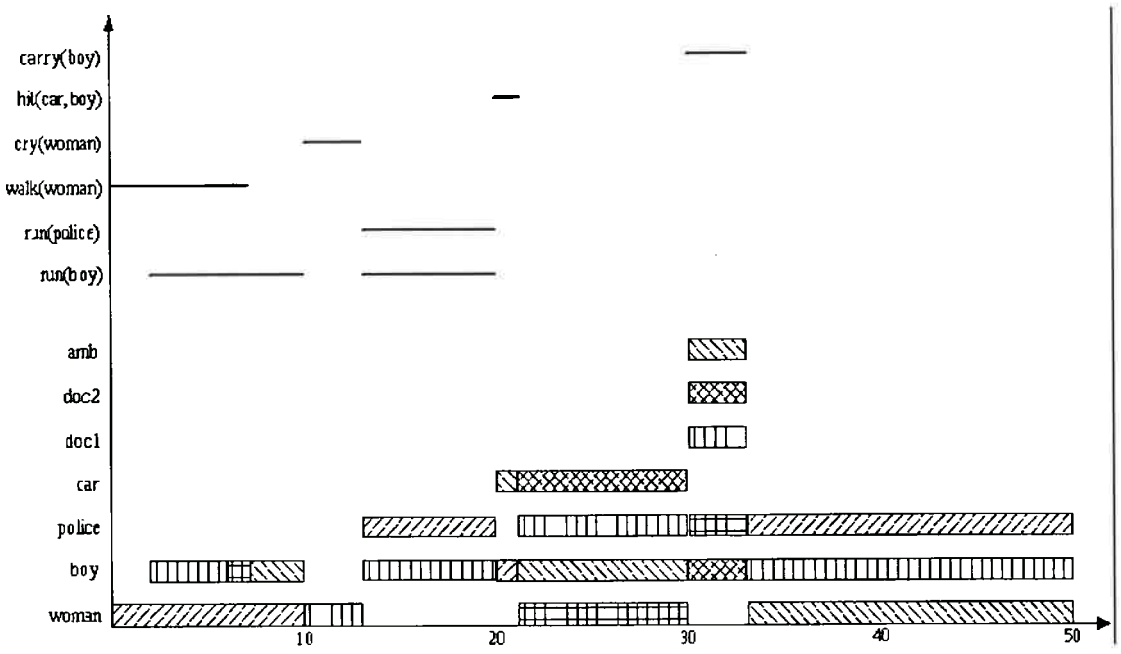


Figura 12 – Modelo ST-AVIS com a subdivisão dos eventos [KÖPRÜLÜ, 04].

O modelo usa um método similar ao MBR para descrever as relações espaciais entre os objetos. Ao invés da área de um retângulo definir um objeto em um quadro, no ST-AVIS, é definida uma área R que corresponde à área do retângulo que o objeto de interesse ocupa durante um intervalo de tempo I, ou seja, durante o intervalo I o objeto pode ser encontrado em qualquer lugar definido pela área R. O problema desta técnica é que quando existem tomadas com zoom ou quando a câmera muda de posição, é preciso redefinir os retângulos o que pode gerar problemas de precisão no relacionamento entre os objetos.

As relações espaciais utilizadas pelo ST-AVIS são as mesmas definidas em [LI (A), 96]. Porém, ao invés de "norte", "sul", "leste", "oeste", utiliza-se simplesmente "em cima", "em baixo", "direita", "esquerda". Para adequar as relações aos movimentos dos objetos ao longo de um intervalo ou a movimentação da câmera, o ST-AVIS trás um novo tipo de relacionamento chamado relacionamento espaço-temporal fuzzy. O relacionamento entre dois objetos A e B durante o intervalo I_k é descrito por $A(\alpha,\mu,I_k)B$, onde α é uma das relações espaciais e μ é o grau de associação. O parâmetro μ é um valor entre [0,1] que representa o quanto um posicionamento entre dois objetos se aproxima de uma das relações espaciais. O valor de μ vai depender da relação espacial e do ângulo entre os baricentros dos retângulos que descrevem os objetos. A forma de calcular o parâmetro é encontrada na Tabela 10.

Tabela 10 – Formas de calcular μ

Relação	**Ângulo ϕ**	**Valor de μ**
Em cima	Arctan(x/y)	$1 - \phi/90^o$
Esquerda	Arctan(y/x)	$\phi/90^o$
Em cima à esquerda	Arctan(x/y)	$1-(abs(\phi-45^o)/45^o)$
Em cima à direita	Arctan(y/x)	$1-((\phi-45^o)/45^o)$

Na Figura 13 é possível ver diferentes graus de associação para o relacionamento "em cima à direita".

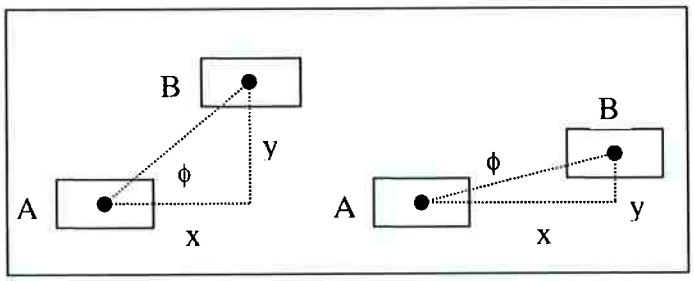
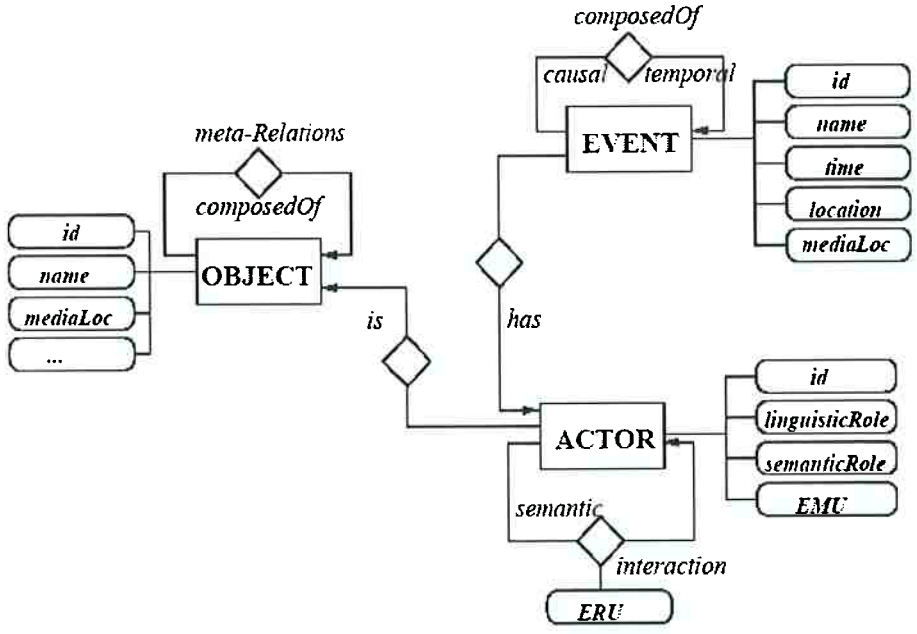


Figura 13 – Relações com diferentes graus de associação.

O modelo consegue responder a solicitações baseadas em regiões, trajetória, relações espaciais e relações espaço-temporais com o conceito fuzzy apresentado.

2.2.6. Modelo Entidade-Relacionamento – Orientado a Objetos.

Em [EKIN, 04] é apresentado um modelo que utiliza conceitos de modelagem entidade-relacionamento e orientação a objetos. O principal objetivo do modelo é descrever eventos em um vídeo. Para atingir tal objetivo, as principais entidades que fazem parte do modelo são os eventos, os objetos que participam dos eventos e os atores que descrevem os papéis dos objetos nos eventos. A definição da entidade "ator" é importante, pois desta forma, é possível descrever com precisão um objeto que em um mesmo vídeo atua em diferentes papéis. Os autores citam como exemplo um jogador de futebol que em um determinado instante marca um gol e em outro passa a bola para um companheiro marcar. O mesmo objeto (jogador) possui papéis diferentes.



Figura' 14 – Representação gráfica das entidades de um evento[EKIN, 04].

Para representar as características de baixo nível, as ações são dividas em EMU, *elementary motion units* e as interações em EMU e ERU, *elementary reaction units*. Estas duas entidades são compostas de quadros e são definidas para cada objeto. O modelo também define três tipos de relações: entre eventos, entre objetos, e

entre atores. A representação gráfica dos objetos, atores, EMU, ERU e seus relacionamentos pode ser vista na Figura' 14. As entidades são representadas por retângulos. Os losangos representam os relacionamentos com os respectivos nomes. As elipses representam os atributos de uma entidade ou relacionamento.

A Figura 15 mostra o funcionamento do modelo para um vídeo de exemplo que contenha a cena de chute em um jogo de futebol. O evento possui dois objetos em seus respectivos papéis (*player* e *ball*).

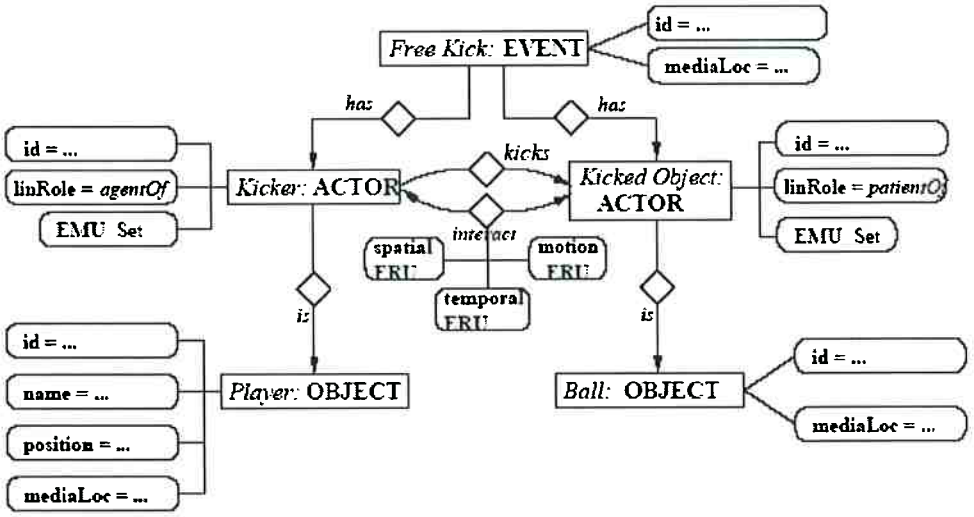


Figura 15 – Diagrama entidade-relacionamento do evento "chute" [EKIN, 04].

Os relacionamento entre jogador e bola estão representados pelas ERUs e EMUs pode ser visto na Figura 16.

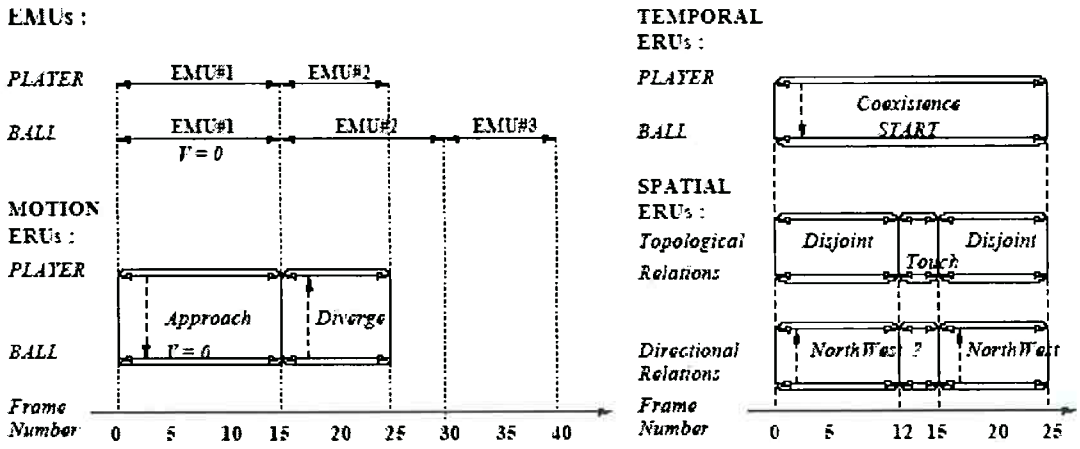


Figura 16 – Representação das EMUs e ERUs e dos Relacionamentos [EKIN, 04].

No lado esquerdo da figura estão listadas as EMUs que descrevem o movimento dos objetos "*Player*"e "*Ball*". As *Motion ERUs* descrevem a aproximação e o afastamento dos objetos. Entre os quadros [15, 25] o movimento entre os objetos é de afastamento. O gráfico do lado direito descreve as relações temporais e espaciais (topológicas e direcionais). Entre os quadros [15, 25], por exemplo, os objetos estão separados (*disjoint*).

O modelo apresenta também o conceito de "*Media Locator*". Todas as entidades citadas acima podem ser encontradas em diferentes intervalos ao longo do tempo. Cada um destes intervalos é armazenado em arquivos de vídeos. Estes arquivos contêm o mesmo evento gravado por diferentes câmeras, em diferentes ângulos, imagens congeladas do evento, quadros chaves, podendo inclusive conter mídias em outros formatos como arquivos texto e arquivos de áudio.

3. MODELO PROPOSTO

3.1. Estruturação das Informações

Em bancos de dados é comum a criação de índices nas tabelas para a otimização das buscas. Geralmente uma tabela é indexada pelas informações que são consultadas com maior freqüência e esta indexação é feita durante a modelagem dos dados. O banco de dados dos clientes de uma empresa, por exemplo, possui uma tabela onde constam informações como nome, endereço e CPF. Um bom índice seria o número do CPF ou o nome do cliente.

No caso de bancos de dados multimídia e em especial bancos de vídeos, as informações contidas nos vídeos (que são os dados em si) estão encapsuladas ao longo do vídeo. Com um arquivo de vídeo em mãos, não é possível saber a priori qual é seu conteúdo, ou seja, qual tipo de informação que ele contém. O vídeo precisa ser modelado e uma camada de estruturação deve ser construída para que a indexação do vídeo seja possível. Esta é a principal diferença entre bancos de dados tradicionais e bancos de dados de vídeos [MARQUES, 04].

A estruturação das informações de um vídeo pode ser vista em [EKIN, 04], [CHEN, 02] e [OOMOTO, 93], onde entidades são criadas de acordo com cada modelo. Em [EKIN, 04] são definidas as entidades "*object*", "*event*", "*actor*" entre outras e a camada de estruturação pode ser vista na Figura' 14. Em [CHEN, 02] são definidas as entidades "*video object*", "*static video object*", "CAI" entre outras e a camada de estruturação destas entidades resulta na Figura 7. Em [OOMOTO, 93] temos as entidades "v*ideo object*", "*interval*" e a camada de estruturação destas entidades resulta na Figura. 3. Além de descrever o modelo de dados, a camada de estruturação possibilita a definição de índices para otimização das buscas. O papel da camada de estruturação em um banco de dados de vídeos pode ser visto na Figura 17.

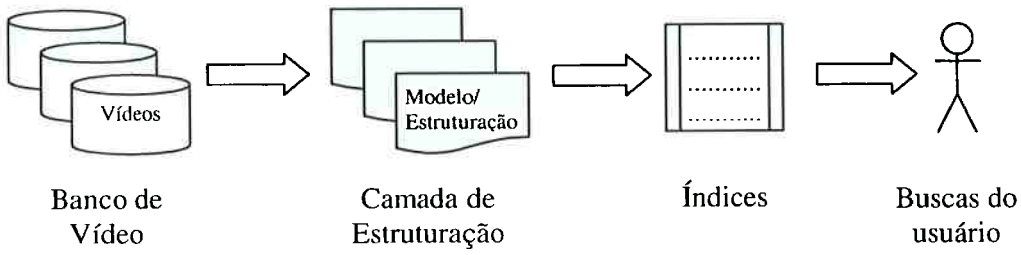


Figura 17 – Estruturação de dados em um banco de dados.

Na Figura 17 a camada de estruturação da informação está destacada em cinza. Os vídeos contidos na base de dados têm suas informações organizadas na camada de estruturação. Esta etapa envolve a segmentação do vídeo em clipes representativos e a abstração do seu conteúdo. Os índices gerados serão utilizados nas consultas feitas pelos usuários.

3.2. Visão Geral.

Como já foi mencionado anteriormente o objetivo deste trabalho é propor uma modelagem de vídeo baseada na semântica de eventos que consiga extrair os segmentos de vídeo que o compõe.

Para atingir este objetivo é preciso definir semanticamente os eventos presentes em um vídeo. Um evento é composto de seqüências de vídeo que aqui serão chamadas de ações semânticas (AS). Estas ações semânticas são representadas no vídeo pelas relações espaço-temporais entre os objetos. O evento deve ser definido semanticamente pelo usuário na forma de ações semânticas para que estas sejam relacionadas pelo modelo com as relações espaciais e relações de existência.

Os objetos e suas relações podem ser identificados para cada quadro. No entanto em um vídeo os objetos e as relações entre eles podem permanecer os mesmos por vários quadros. Desta forma, os quadros com propriedades em comum podem ser agrupados em clipes. O primeiro quadro de cada clipe será chamado de quadro-chave. Cada um destes clipes possui um significado semântico e serão chamados de Clipes Semânticos (CS). Cada clipe semântico corresponde a um conjunto de quadros onde as relações entre os objetos de interesse se conservam.

Uma seqüência de clipes semânticos pode vir a representar uma relação espaço-temporal e conseqüentemente uma ação semântica.

A estrutura hierárquica do modelo proposto pode ser vista na Figura 18.

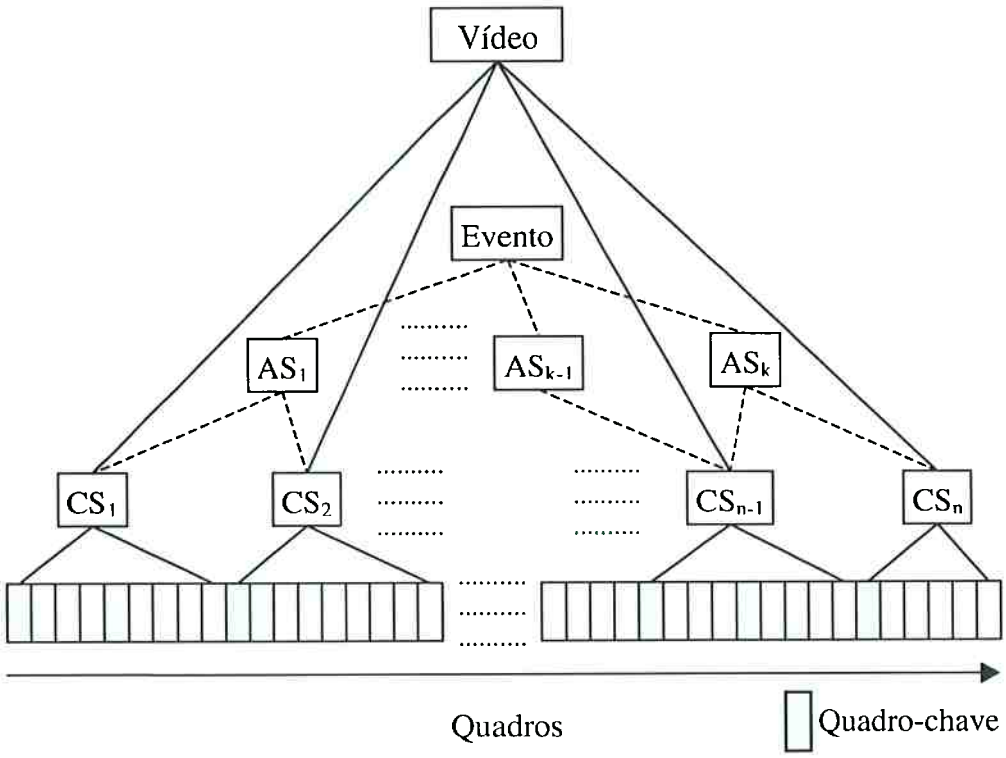


Figura 18 – Estrutura hierárquica do modelo proposto.

Na Figura 18 as linha cheias representam a hierarquia da camada "física" de um vídeo. A camada recebe este nome porque qualquer vídeo objetos que se relacionam entre si. Portanto, os clipes semânticos são elementos de todos os vídeos. Já as linhas tracejadas representam a hierarquia da camada "abstrata" do vídeo. A abstração de clipes semânticos em ações semânticas e das ações em eventos pode ou não ocorrer dependendo do conteúdo dos vídeos.

O reconhecimento de objetos e a identificação das relações espaciais são considerados neste trabalho como sendo tarefa do processamento de imagem. O modelo proposto é independente da capacidade do processamento de imagem de obter objetos e atributos podendo, inclusive, ser feito manualmente.

Juntamente com os objetos e suas relações, duas informações são fundamentais para caracterizar um evento: o local onde o evento ocorreu e quando o

evento ocorreu. Além da dificuldade de se identificar essas informações no conteúdo dos vídeos, pode-se deparar com a situação onde as informações simplesmente não estão presentes no vídeo.

O sistema possui quatro tipos de atores: O modelo, o processamento de imagens, o modelador de eventos e o usuário final. As definição de responsabilidades dos atores em cada etapa do sistema pode ser vista na Figura 19

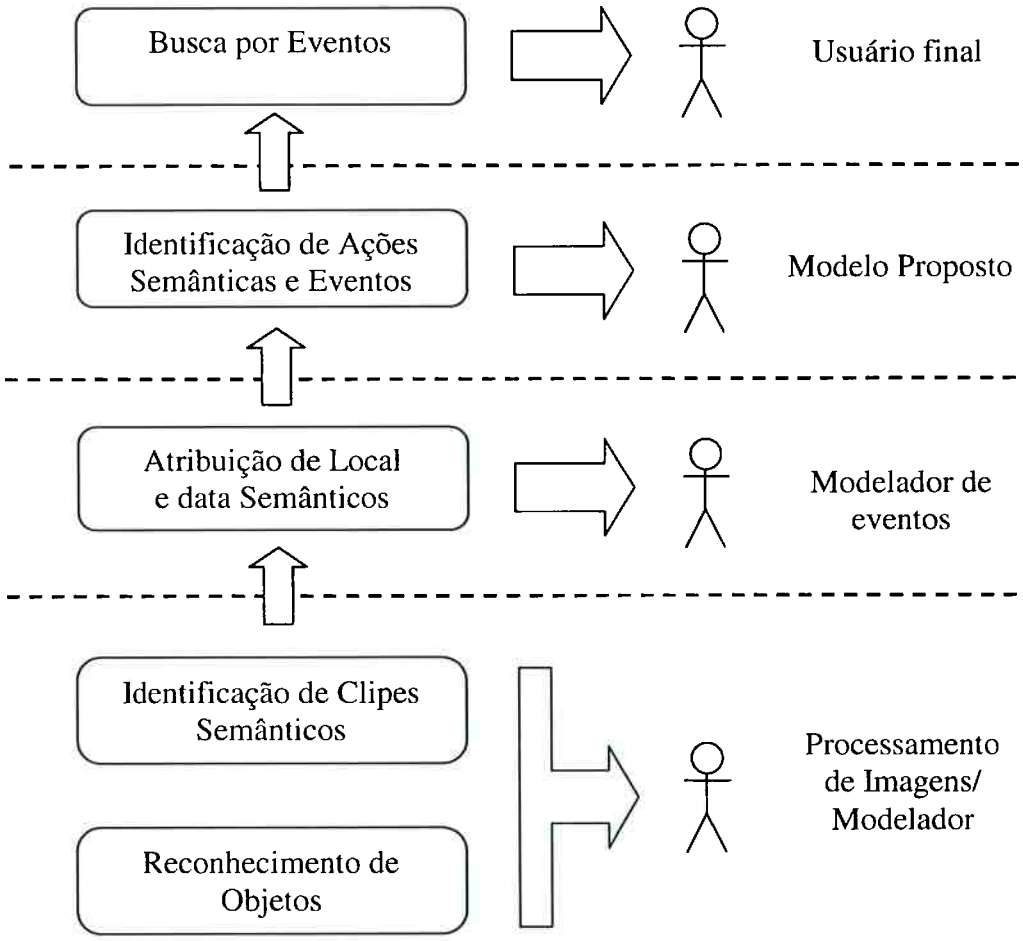


Figura 19 – Atividades e responsabilidades na indexação de um vídeo.

O modelo possui a estrutura que permite o armazenamento da base de conhecimento gerada pelo cadastramento de eventos e vídeos e desta forma atende as solicitações realizadas pelo usuário final. O modelo identifica as ações semânticas e os eventos previamente descritos. O processamento de imagens é responsável pelo reconhecimento de objetos, identificação das relações espaciais e temporais entre eles e determinação dos clipes semânticos. Este processo pode ser automático ou

manual. O modelador de eventos terá que descrever o evento em ações semânticas para alimentar a base de eventos do modelo. O modelo proposto conta ainda com o modelador de eventos para realizar a atribuição do local e data de um evento que serão chamados de Local Semântico (LocalS) e Data Semântica (DataS). Para um vídeo de um jogo de futebol, por exemplo, o local semântico é o "Estádio do Maracanã" e a data semântica é "14 de Janeiro de 2000 às 20:00". O conteúdo de um vídeo pode se passar em vários lugares e em diferentes datas. Desta forma a atribuição deve ser feita por clipe semântico e o vídeo herda esses atributos como será visto mais adiante.

3.3. Definições do Modelo.

A seguir as entidades que compõe o modelo são descritas em detalhe.

A. Quadros: Os quadros são as entidades fundamentais de um vídeo. Em cada quadro são extraídos os objetos de vídeo e são determinadas as relações entre eles. Formalmente um quadro é descrito por $F_i = \{F_{id}, O_L\}$ onde F_{id} é o identificador do quadro e O_L a lista de objetos contidos no quadro.

B. Objeto de Vídeo: Objetos de Vídeo são objetos físicos extraídos de um vídeo. Cada objeto de vídeo irá instanciar um objeto físico. Este objeto pode ser uma pessoa, um carro, uma cadeira. Como são representados diferentes tipos de objetos, cada tipo possuirá uma classe própria e seus próprios atributos. Todas essas classes herdam as características da superclasse objetos de vídeo. Formalmente um objeto é representado por $O_j = \{Oid, T, (a, v)\}$ onde Oid é o identificador do objeto, T é o tipo do objeto e (a, v) é um conjunto de pares atributo/valor. Um exemplo de diagrama de classes para os objetos "Carro"e "Pessoa" pode ser visto na Figura 20.

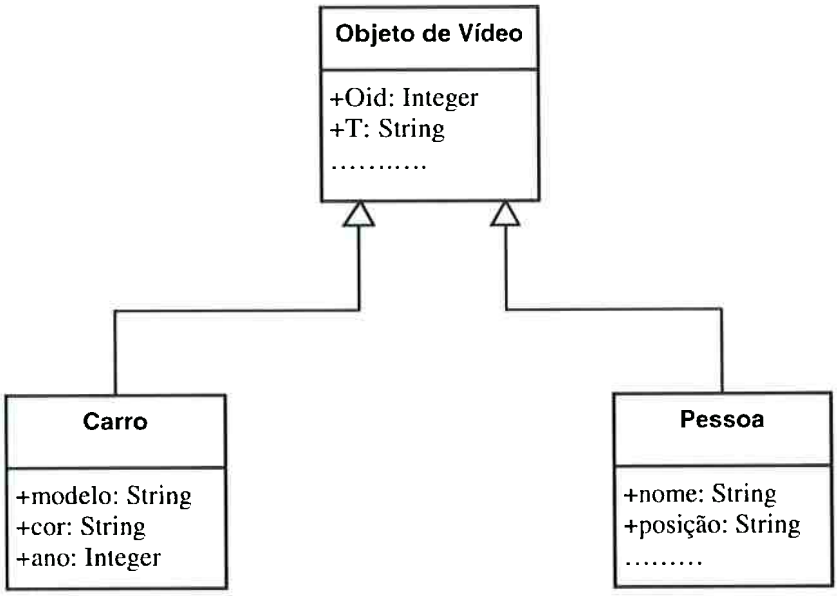


Figura 20 – Exemplo de diagrama de classe de objetos de vídeo.

Os objetos dos tipos (T) "Carro" e "Pessoa" possuem atributos (a, v) próprio como, por exemplo, modelo e posição respectivamente. No entanto os dois tipos de objetos herdam os atributos "Oid" e "T".

O relacionamento entre os objeto bem como seus atributos são determinados pelo processamento de imagem. Quanto maior for a eficiência do processamento, maior será a complexidade das solicitações que poderão ser atendidas. Por exemplo, se o processamento conseguir identificar pessoas, solicitações do tipo "Retorne todos os vídeos onde determinada pessoa entra na sala" serão atendidas. Porém, se o processamento conseguir extrair também atributos como cor de roupas, solicitações do tipo "Retorne todos os vídeos onde determinada pessoa entra na sala com uma camisa azul" poderão se atendidas.

C. Clipe Semântico: Clipes semânticos são seqüências de quadros que descrevem uma relação entre os objetos de vídeo. Um clipe semântico é definido quando os objetos se mantêm em cena e suas relações se mantêm inalteradas. Formalmente um clipe semântico é definido por $CS_g = \{CSid, F_L, L_S, D_S, r_{ij}\}$ onde CSid é o identificador do clipe semântico, F_L é uma seqüência de quadros onde as condições citadas anteriormente são verdadeiras, r_{ij} é a relação espacial entre os objetos O_i e O_j,

L_S é o local semântico e D_S é a data semântica. O primeiro quadro do clipe semântico é chamado de quadro chave. É o primeiro quadro onde as condições de existência de um CS são verdadeiras. Como os CS estão ordenados, pode-se dizer que os quadros-chave determinam as fronteiras dos CS.

As Relações Temporais consideradas pelo modelo utilizam as relações definidas em [ALLEN, 83] e relacionadas na Tabela 1.

As relações espaciais utilizadas no modelo são as baseadas nas relações definidas em [LI (A), 96] e relacionadas na Tabela 3. Como em um clipe podem existir vários objetos as relações espaciais são formalmente descritas por r_{ij} = {rid, rel, O_i, O_j} onde rid é um nome que identifica a relação espacial, rel é o predicado da relação espacial e O_i e O_j são objetos de vídeo.

D. Operador de Seqüência: Para relacionar os clipes semânticos será utilizado o operador de seqüência "→" que descreve tanto algebricamente como graficamente a seqüência esperada de clipes semânticos. Desta forma, a expressão $CS_1 \rightarrow CS_2$ indica que CS_1 ocorre antes de CS_2. O operador de seqüência será utilizado também para relacionar as ações semânticas como será visto a seguir.

E. Ação Semântica: Ações Semânticas são seqüências de Clipes Semânticos que descrevem uma relação espaço-temporal ou uma relação de existência. Ações Semânticas são representadas graficamente com o auxilio de grafos. Um grafo G é definido por G(V, A) onde V é um elemento do conjunto de vértices ou nós e A é um elemento do conjunto de conectores do grafo. Cada vértice representará uma ação semântica. Sendo assim:

$$V = \{AS_i \mid AS_i \text{ é uma ação semântica de um vídeo}\}.$$
$$A = \{(AS_i, AS_{i+1}) \mid A_i \text{ ocorre antes de } AS_{i+1}\}.$$

Um exemplo de grafo pode ser visto na Figura 23.

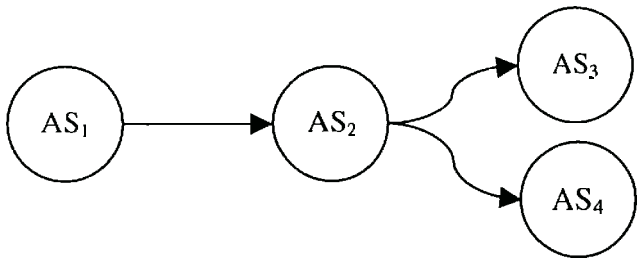


Figura 21 – Exemplo de grafo orientado.

O grafo da Figura 21 é definido por:

$$V = \{AS_1, AS_2, AS_3, AS_4\}.$$
$$A = \{(AS_1, AS_2),(AS_2, AS_3),(AS_2, AS_4)\}.$$

Formalmente uma ação semântica é definida por $AS_m = \{AS_{id}, CS_G, R_{ij}\}$ onde AS_{id} é um identificador da ação semântica, CS_G é um grafo de Clipes Semânticos e R_{ij} é a relação espaço temporal entre os objetos O_i e O_j. Um exemplo de ação semântica é visto na Figura 22.

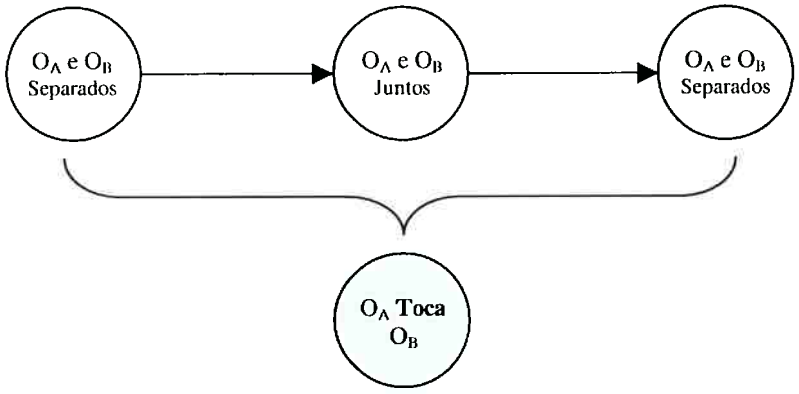


Figura 22 – Exemplo da Ação Semântica "Tocar".

De acordo com a Tabela 4, a descrição algébrica da relação "Tocar" é:

$$dj \rightarrow mt \rightarrow dj$$

Ou seja, para se obter a relação "tocar", é preciso que dois objetos apresentem entre si as relações "separados"(dj), "juntos"(mt), "separados"(dj). Cada uma destas relações é um clipe semântico.

As relações espaço-temporais são constituídas por uma seqüência de relações espaciais orientadas no tempo. As relações espaço-temporais definem uma ação de interação realizada entre os objetos. As relações espaço-temporal utilizadas no modelo são as encontradas em [ERWIG, 99] e relacionadas na Tabela 4. Formalmente uma relação espaço-temporal é descrita por $R = \{R_{id}, R_n, r_t, t\}$ onde R_{id} é um identificador da relação entre dois objetos de vídeo, R_n é nome da relação e r_t é a seqüência de relações espaciais ordenada no intervalo de quadros t. Formalmente t é descrito por $t = [f_i, f_j]$ onde f_i e f_j correspondem ao quadro inicial e final respectivamente. As relações de existência dos objetos nos clipes semânticos são as encontradas em [CHEN, 04] e relacionadas na Tabela 6.

F. Evento: Eventos são conjuntos de ações semânticas, objetos e relações entre os objetos que estão contidos em um vídeo. A definição semântica de um evento é feita baseada na percepção humana, ou seja, cabe a um modelador a descrição do evento em ações semânticas que juntas formam um grafo com a seqüência lógica. A esta seqüência de ações semânticas é atribuído um nome que descreve o evento. Os eventos são representados graficamente com o auxilio de grafos. Um evento é formalmente descrito por $E = \{E_{id}, AS_G\}$, onde E_{id} é o identificador que descreve semanticamente o evento e AS_G um grafo orientado de ações semânticas. Em um grafo representando um evento nem sempre todas as ações ocorrem. Isso ocorre porque diferentes seqüências de ações representam o mesmo evento O caminho percorrido no grafo pode variar de acordo com as ações que ocorrem no vídeo.

Um exemplo de evento pode ser encontrado na Figura 23.

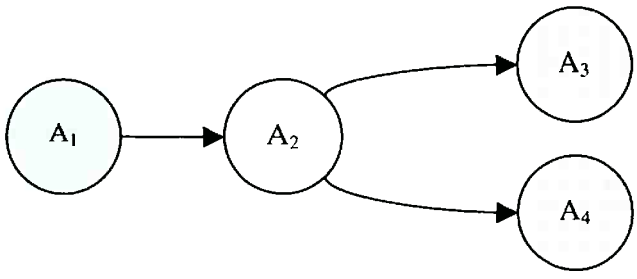


Figura 23 – Exemplo de grafo orientado representado evento E.

O nó cinza indica qual o nó de início do evento e o nó quadriculado indica qual o nó e término. O evento E existe quando ocorrerem as ações semânticas A1 → A2 → A3 ou A1→ A2 → A4. O operador "→" além de ser o elo entre as ações também indica que há uma ordem entre elas, seja, A1 → A2 é semanticamente diferente de A2 → A1.

O grafo do evento pode possuir múltiplos nós de início e múltiplos nós de término dependendo da lógica de seqüência. A lógica de seqüência pode ser melhor compreendida com o auxílio dos exemplos de blocos da Figura 24.

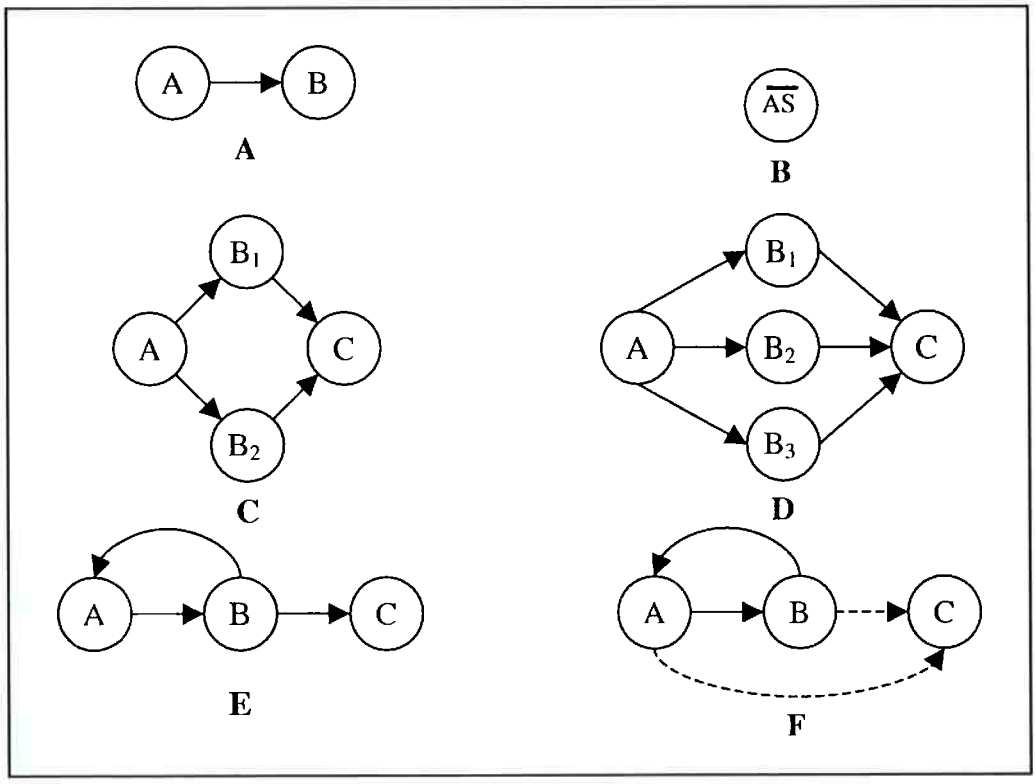


Figura 24 – Blocos utilizados nos grafos.

O exemplo A traz uma seqüência simples entre AS. O exemplo B é uma ação negada. Uma ação semântica com uma barra em cima representa uma ação negada, ou seja, quando a ação representada não ocorre mais até o final do vídeo. Este tipo de nó é importante para a determinação do fim de eventos que ocorrem em partes, por exemplo, eventos esportivos (que possuem sets, quartos e tempos) e entrevistas (que são realizadas em blocos). O exemplo C mostra um ponto de decisão onde o evento pode seguir dois caminhos distintos. Ao chegar à ação A, o grafo pode

prosseguir pela ação B_1 ou pela ação B_2. E neste bloco em especial, os dois caminhos levam a ação C. O exemplo D mostra um ponto de decisão onde o número de possibilidades é maior que no exemplo anterior. O exemplo E representa uma ou mais ações que se repetem um certo número de vezes antes de prosseguir. As ações A e B se repetem um certo número de vezes até que o grafo prossiga para a ação C. O exemplo F traz uma lógica onde o evento pode seguir para uma nova AS ou repetir as AS que já ocorreram. As ações A e B pode se repetir por um certo número de vezes ou o grafo pode seguir a seqüência $A \rightarrow C$.

O conector tracejado que pode ser visto no bloco "F" indica que a ação semântica apontada pelo arco precisa ocorrer para o evento ser identificado, mas não faz parte do evento. É apenas uma condição de existência.

G. Trecho de Vídeo: Um trecho de vídeo é a concretização de um evento. Quando procuramos por um evento em um vídeo, o que o modelo retorna são trechos de vídeos onde este evento ocorre.

H. Vídeo: Um vídeo é uma mídia que consiste em uma seqüência de clipes. Cada clipe é a representação de uma cena. Cada cena é constituída de quadros. Para cada vídeo, são extraídos certos objetos de interesse. Formalmente um vídeo é representado por $V_m = \{Vídeo_{id}, CS_L, LocalS_L, DataS_L\}$ onde $Vídeo_{id}$ é o identificador do vídeo, CS_L é a lista de Clipes Semânticos que compõe o vídeo, $LocalS_L$ é uma coleção de locais semânticos em que se passam os CS e $DataS_L$ é uma coleção de datas semânticas de quando se passam os eventos do vídeo. Os elementos destas duas coleções são herdados dos clipes semânticos e podem possuir o mesmo valor para todo o vídeo.

3.4. Algoritmo de Indexação e Motor de Inferência.

O modelo proposto é composto de uma base conhecimento que possui um cadastro com as regras para definição de cada evento (representadas pelos grafos). Esta base de conhecimento armazena um conjunto de regras para cada

evento, suas respectivas ações e clipes semânticos e é consultada durante a indexação dos eventos de um vídeo.

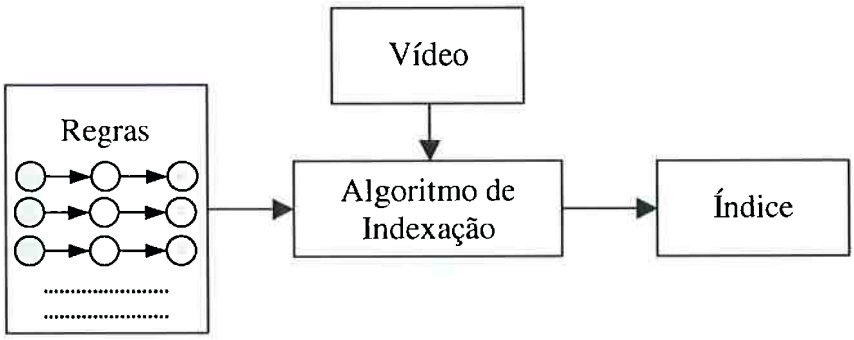


Figura 25 – Estrutura da base de conhecimento.

Cada evento da base de conhecimento possui relacionamentos com as AS que o descrevem e as AS possuem relacionamentos com os CS que as descrevem. Um evento possui uma ou mais AS e uma AS pode estar presente em um ou mais eventos. Por isso o relacionamento possui multiplicidade (n..n). O mesmo se aplica ao relacionamento entre as AS e os CS.

Da mesma forma que em um sistema *Prolog* [PALAZZO, 97], o modelo proposto aceita os fatos (clipes semânticos) e as regras (ações semânticas) como um conjunto de axiomas e a adição de um evento como um teorema a ser provado. A identificação de eventos de um vídeo a partir da base de conhecimento funciona como um motor de inferência. Cada evento equivale a uma regra do motor de inferência. Para o cadastramento de um vídeo a lista de quadros chaves é percorrida com o intuito de encontrar uma seqüência idêntica a um dos caminhos do grafo que descreve o evento. Quando a busca está no meio de uma possível seqüência e a validação de quadros-chave falha, o mecanismo de *backtracking* é acionado fazendo com que a busca retorne pelo mesmo caminho e tente uma solução alternativa o que equivale a buscar por outro caminho no grafo.

O modelo proposto aplicado a uma coleção de vídeos gera um conjunto de dados que alimenta a camada de estruturação do modelo. A Figura 26 está representando a camada de estruturação destes dados. A classe "evento" armazena todos os eventos que foram encontrados nos vídeos. A classe "vídeo" armazena informações dos vídeos que já foram indexados pelo sistema. A classe

"trecho_de_vídeo" armazena as referências para os trechos de vídeo onde ocorrem os eventos. A classe "quadro_chave" contém as informações que descrevem os clipes semânticos dos vídeos. Como foi visto anteriormente, o modelador cadastra o local e data que o clipe semântico ocorre. Estas informações são armazenadas nas classes "local_semantico" e "data_semantica". A classe "objeto" armazena os objetos de vídeo que foram identificados nos vídeos. Os objetos podem ser reconhecidos pelo sistema de processamento de imagem ou cadastrados também pelo modelador. A classe "atributos" armazena os atributos que podem ser encontrados nos objetos de vídeo.

Da forma como foi modelado, o sistema funciona como um sistema especialista. Para que o usuário possa ter acesso ao processo de raciocínio que o sistema utilizou, desenvolveu-se a entidade "explicação" que armazena, para cada evento identificado em um vídeo, qual a regra e caminho utilizados. Este meta-conhecimento pode ser de grande utilidade para se verificar a precisão dos resultados e eventualmente validá-los. Esta necessidade de explicação de como o sistema identificou o evento é fundamental em áreas críticas como, por exemplo, diagnósticos médicos.

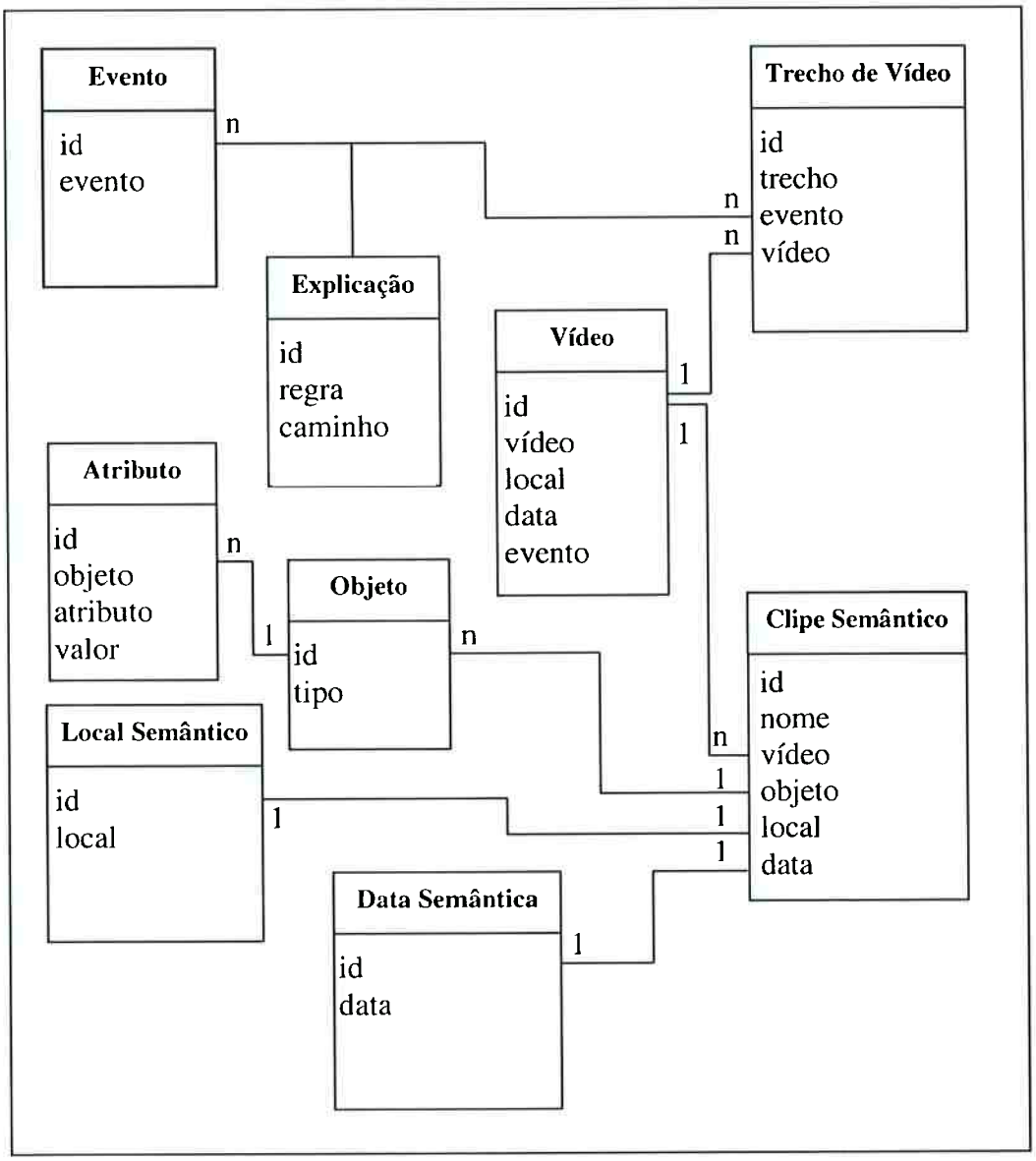


Figura 26 – Representação da camada de estruturação dos dados.

Uma vez estruturadas, o relacionamento entre estas classes gera um índice composto. Este índice irá permitir buscas mais eficientes na base de dados.

Com a base de conhecimento preenchida com as regras dos eventos é possível aplicá-la para a identificação dos eventos nos vídeos. Nesta tarefa, cada parte do evento deve ser encontrado no vídeo. O algoritmo que busca eventos nos vídeos pode ser visto na Figure 27.

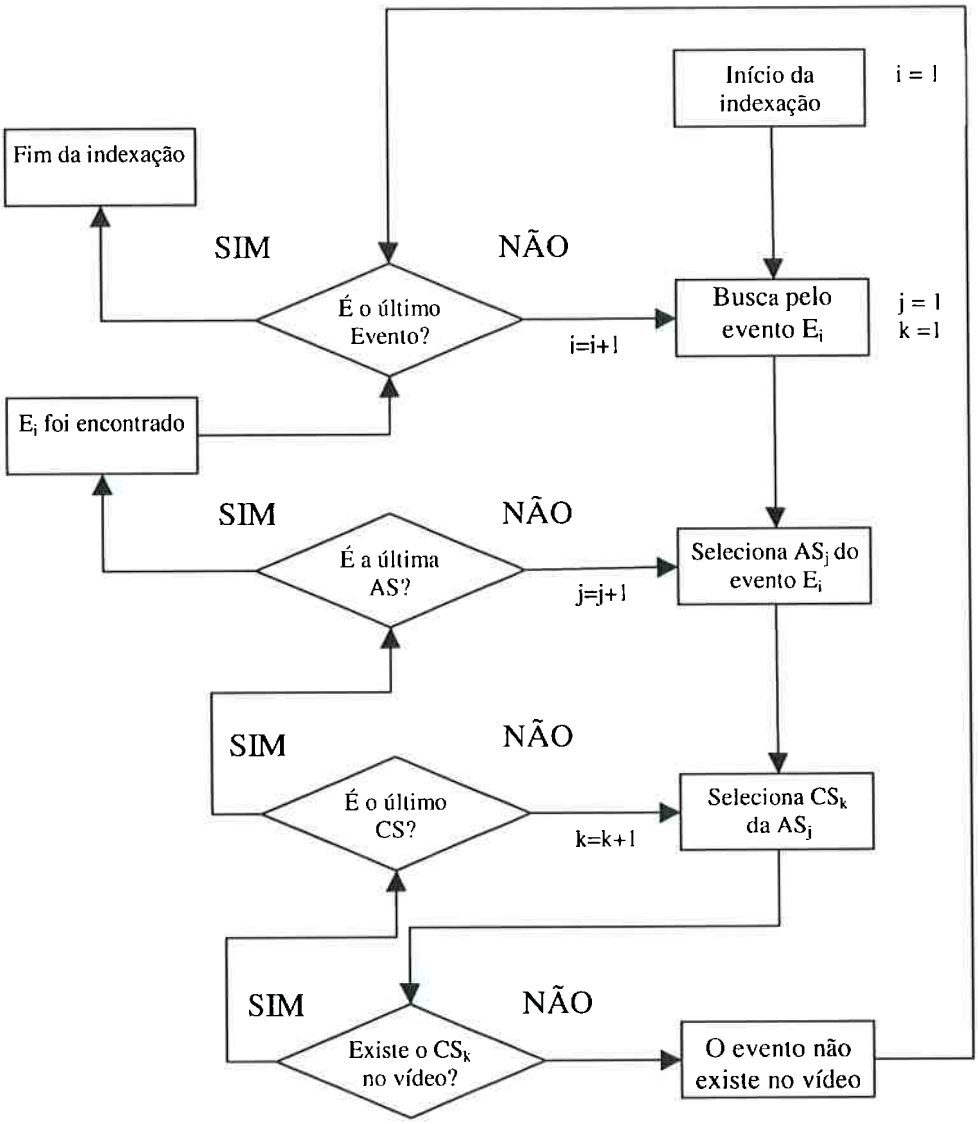


Figure 27 – Algoritmo de indexação de eventos.

Embora não seja escopo deste trabalho definir uma linguagem de busca das informações nos vídeos, é possível utilizar SQL para exemplificar consultas à base de dados. A solicitação de um usuário, por exemplo, "Retorne todos os vídeos onde aparecem gols do Pelé" será uma consulta SQL transparente para o usuário como pode ser visto na Figura 28:

```
SELECT tv.trecho
FROM evento e,
     video v,
     trecho_video tv,
     clipe_semantico cs,
     objeto o,
     atributo a,
WHERE e.evento = "gol"
AND   v.evento = e.id
AND   tv.video = v.id
AND   cs.trecho_video = tv.id
AND   cs.objeto =o.id
AND   o.tipo = pessoa
AND   o.id = a.objeto
AND   a.atributo = jogador
AND   a.valor = pelé
```

Figura 28 – Query para a busca do evento "gol".

4. ESTUDO DE CASO

Nesta seção o modelo proposto será aplicado em três situações diferentes: vídeos de esportes, em especial jogos de futebol onde o objetivo é a extração de trechos de vídeo com o evento "gol", vídeos com programação de TV onde o objetivo é a extração de trechos de vídeo onde ocorram o evento "entrevista" e vídeos de segurança onde o objetivo é a extração de trechos de vídeo com o evento "furto". No exemplo do evento "furto" será aplicada a álgebra definida nas seção 3.3.

A. Evento "gol".

O primeiro passo é o levantamento dos objetos de vídeo envolvidos no evento. No caso do evento "gol" (E_{gol}) os objetos envolvidos são:

- Jogador - $O_{jogador}$.
- Bola - O_{bola} .
- Gol - O_{trave}.

O segundo passo é identificar as ações semânticas que compõe o evento e descrevê-las em função das relações espaço-temporais e das relações de existência dos objetos de vídeo. O E_{gol} pode ser descrito pela seguinte seqüência de ações semânticas:

- **Jogador recebe a bola:**

A ação semântica "Jogador recebe a bola" pode ser descrita como O_{bola} "Pega" $O_{jogador}$. O predicado "Pega" representa a seguinte relação espaço temporal:

$$AS_{recebe} \Rightarrow CS_{separados} \rightarrow CS_{juntos}$$

onde $CS_{separados}$ representa a relação "O_{bola} e $O_{jogador}$ *Separados*" e CS_{juntos} representa a relação "O_{bola} e $O_{jogador}$ *Juntos*".

- **Jogador aciona a bola:**

A ação semântica "Jogador aciona a bola", é descrita então por $O_{jogador}$ "Solta" O_{bola}. O predicado "Solta" representa a seguinte relação espaço temporal:

$$AS_{solta} \Rightarrow CS_{juntos} \rightarrow CS_{separados}$$

onde CS_{juntos} representa a relação "O_{bola} e $O_{jogador}$ *Juntos*" e $CS_{separados}$ representa a relação "O_{bola} e $O_{jogador}$ *Separados*".

- **Bola entra no gol:**

A ação semântica "Bola entra no gol" é descrita por O_{bola} "Entra" O_{trave}. O predicado "Entra" representa a seguinte relação espaço temporal:

$$AS_{entra} \Rightarrow CS_{separados} \rightarrow CS_{juntos} \rightarrow CS_{sobrepostos} \rightarrow CS_{cobre} \rightarrow CS_{dentro}$$

onde $CS_{separados}$ representa a relação "O_{bola} e O_{gol} *Separados*", CS_{juntos} representa a relação "O_{bola} e O_{gol} *Juntos*", $CS_{sobrepostos}$ representa a relação "O_{bola} e O_{gol} *Sobrepostos*", CS_{cobre} representa a relação "O_{trave} *Cobre* O_{bola}" e CS_{dentro} representa a relação "O_{bola} *Dentro* O_{trave}".

Pode-se então definir o grafo que representa o evento gol como pode ser visto na Figura 29.

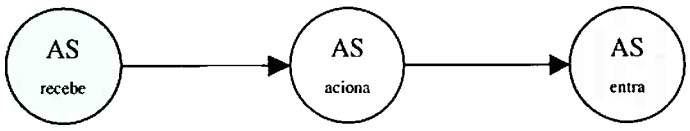


Figura 29 – Grafo representando o evento "gol"

O evento "gol" pode então ser descrito por:

$$E_{gol} = \{Eid, (AS_{recebe}, AS_{aciona}, AS_{entra})\}.$$

Com a definição do evento "gol", a seguinte solicitação pode ser feita: "Retorne todos os trechos de vídeo que contenham um gol".

Uma possível solução desta solicitação pode ser vistos na Figura 30.

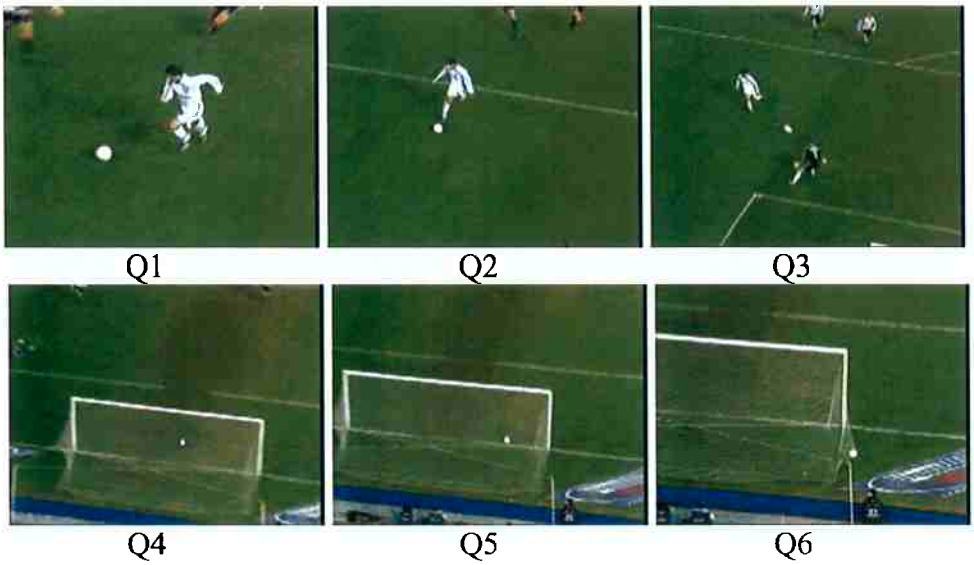


Figura 30 – Seqüência de quadros do evento "gol"[EKIN, 04].

Os quadros Q1 e Q2 representam a ação semântica do jogador recebendo a bola. O quadro Q3 representa a ação do jogador acionando a bola e os quadros Q4, Q5 e Q6 representam a ação da bola entrando no gol.

Da forma como foi modelado o evento "gol" retorna clipes de vídeo que começam instantes antes do jogador que faz o gol receber a bola. No entanto, se o evento "gol" contemplar a troca de passe que originou o gol, a modelagem deve ser alterada. Deve-se incluir a ação semântica "passe" no inicio da modelagem. Por outro lado o gol pode ser originado de uma falta, ou seja, sem troca de passes. Percebe-se então que existem diferentes possibilidades de se iniciar os eventos. A Figura 31 trás o grafo do evento E_{gol} modificado.

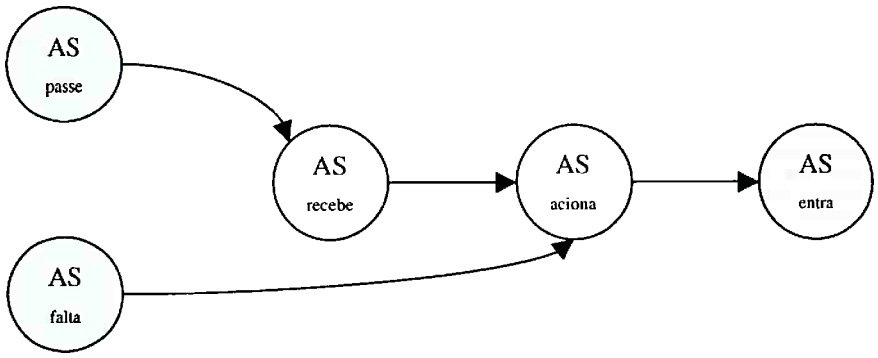


Figura 31 – Grafo do Evento "gol" modificado.

B. Evento "furto".

Em aplicações de segurança é comum a utilização de câmeras para monitorar o acesso de pessoas em ambientes. Os vídeos obtidos com estas observações podem ser utilizados na identificação dos envolvidos em furtos e roubos. Se imaginarmos a situação onde um objeto desaparece de um ambiente, pode-se aplicar o modelo para identificar o trecho de vídeo onde o evento "furto do objeto" ocorreu. O primeiro passo é o levantamento dos objetos de vídeo envolvidos no evento. No caso do evento E_{furto} os objeto de vídeos envolvidos são:

- Pessoa - $O_{pessoa} = \{O_1, (pessoa, funcionário)\}$.
- Caixa - $O_{caixa} = \{O_2, (objeto, caixa)\}$.

O segundo passo é identificar as ações semânticas que compõe o evento e descrevê-las em função das relações espaço-temporais e das relações de existência dos objetos de vídeo. O evento furto pode então possuir as seguintes AS:

- **Pessoa entra na sala:**

A ação semântica "Pessoa entra na sala" representa a transição de quadros onde a pessoa aparece na sala. A ação pode ser descrita através da seguinte relação de existência: "O_{pessoa} *Aparece* na sala". O predicado "Aparece" representa a seguinte seqüência de relações:

$$CS_{não\ existe} \rightarrow CS_{existe}$$

Onde $CS_{não\_existe}$ representa a relação "O_{pessoa} não existe no clipe" e CS_{existe} representa a relação "O_{pessoa} existe no clipe".

Desta forma, a $AS_{aparece}$ pode ser escrita da seguinte forma:

$$AS_{aparece} = \{AS_1, (CS_{existe}, CS_{não\_existe}), \text{"aparece"}\}.$$

- **Pessoa pega a caixa:**

A ação semântica "Pessoa pega a caixa" representa a transição de quadros onde a pessoa se desloca de encontro à caixa. A ação pode ser descrita através da seguinte relação de existência: "O_{pessoa} *Pega* O_{caixa}". O predicado "Pega" representa a seguinte seqüência de relações:

$$CS_{separado} \rightarrow CS_{junto}$$

Onde $CS_{separado}$ representa a relação O_{pessoa} e O_{caixa} separados e CS_{juntos} representa a relação "O_{pessoa} e O_{caixa} juntos".

Desta forma, a AS_{pega} pode ser escrita da seguinte forma:

$$AS_{pega} = \{AS_2, (CS_{separado}, CS_{junto}), \text{"pega"}\}.$$

- **Pessoa e caixa deixam a sala:**

A ação semântica "Pessoa e caixa deixam a sala" representa a transição de quadros onde a pessoa deixa a sala levando consigo a caixa. A ação pode ser descrita através da seguinte relação de existência: "O_{pessoa} e O_{caixa} *Desaparecem* da sala". O predicado "Desaparecem" representa a seguinte seqüência de relações:

$$CS_{existe} \rightarrow CS_{não\_existe}$$

Onde CS_{existe} representa a relação O_{pessoa} existe no clipe e $CS_{não\_existe}$ representa a relação O_{pessoa} e O_{caixa} não existe no clipe.

Desta forma, a $AS_{desaparece}$ pode ser escrita da seguinte forma:

$$AS_{desaparecde} = \{AS_3, (CS_{existe}, CS_{não\_existe}), \text{"desaparece"}\}.$$

Pode-se então definir o grafo que representa o evento "furto" como pode ser visto na Figura 32.

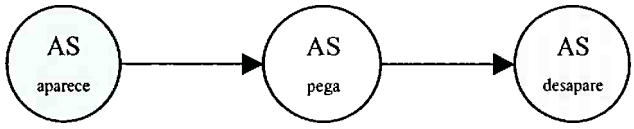


Figura 32 – Grafo representando o evento "furto".

O evento "furto" pode então ser descrito por:

$$E_{furto} = \{E_{id}, (AS_{aparece}, AS_{pega}, AS_{desaparece})\}.$$

Com a definição do evento "furto", a regra que o define pode ser aplicada a uma coleção de vídeos. A Figura 33 mostra um trecho de um vídeo de segurança onde é possível ver uma pessoa em uma atividade suspeita que pode ser classificada como furto de acordo com a regra definida anteriormente.

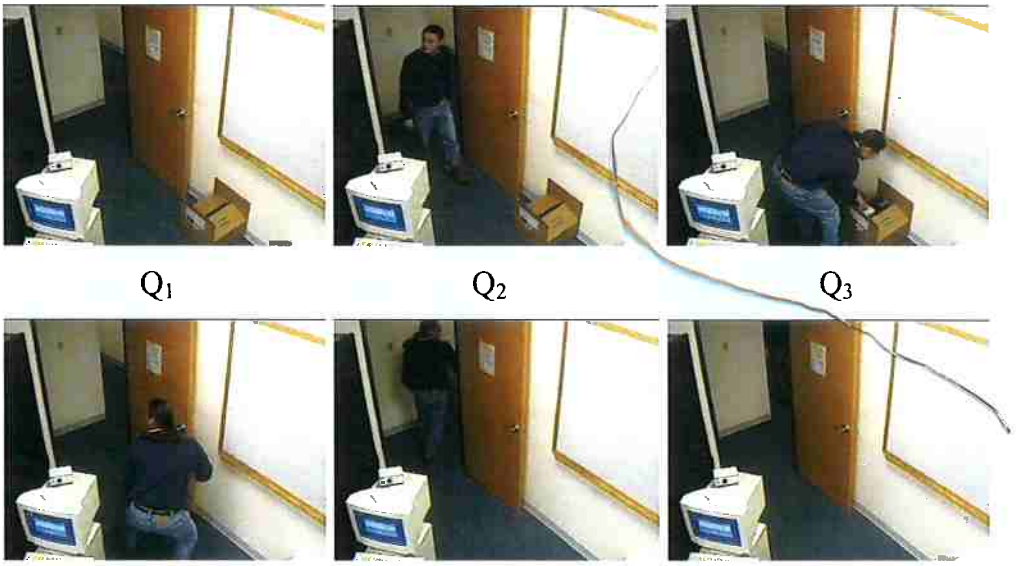


Figura 33 – Seqüência de quadros do evento "furto".

A ação semântica $AS_{aparece} \Rightarrow CS_{não\_existe} \rightarrow CS_{existe}$ é identificada pelos quadros Q_1 e Q_2. O quadro Q_1 mostra a sala com a caixa antes da entrada da pessoa. O quadro Q_2 mostra o instante que a pessoa entra na sala. Desta forma pode-se definir:

$CS_{não\_existe}$= {CS_1, [Q_i, Q_j],“ Recepção”, “13/12/2004 12:40”, “não existe”}

CS_{existe}= {CS_2, [Q_k, Q_l], "Recepção", "13/12/2004 12:40", "existe"}.

A ação semântica $AS_{pega} \Rightarrow CS_{separados} \rightarrow CS_{juntos}$ é identificada pelos quadros Q_2 e Q_3. O quadro Q_3 flagra o instante que a pessoa se aproxima da caixa para verificar seu conteúdo. Desta forma pode-se definir:

$CS_{separados}$={CS_3, [Q_m, Q_n], "Recepção", "13/12/2004 12:40", "separados"}
CS_{juntos} ={CS_4, [Q_o, Q_p]," Recepção", "13/12/2004 12:40", "juntos"}

A ação semântica $AS_{desaparece} \Rightarrow CS_{existe} \rightarrow CS_{não\_existe}$ é identificada pelos quadros Q_4, Q_5 e Q_6. Os quadros Q_4 e Q_5 mostram a pessoa pegando a caixa e deixando a sala. O quadro Q_6 mostra finalmente a sala sem a pessoa e sem a caixa após o furto.

CS_{existe}= {CS_5, [Q_q, Q_r], "Recepção", "13/12/2004 12:40", "existe"}.
$CS_{não\_existe}$={CS_6, [Q_s, Q_t]," Recepção", "13/12/2004 12:40", "não existe"}

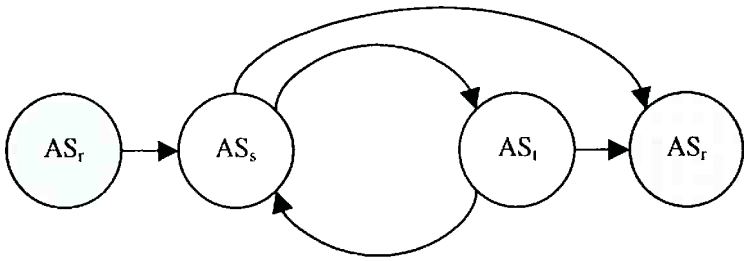
Após a indexação, a seguinte solicitação pode ser feita: "Retorne todos os trechos de vídeo onde ocorre o evento furto". Uma solução para esta solicitação seria o trecho de vídeo da Figura 33.

C. Evento "entrevista".

Como o modelo é baseado na observação de relações entre os objetos, vídeos com *closes* ou alternância de câmeras podem dificultar ou até mesmo impedir a identificação de um evento. Um exemplo onde se pode deparar com estas limitações é a busca pelo evento "entrevista". Uma entrevista pode possuir três tipos enquadramentos dos objetos de interesse, entrevistador e entrevistado:

- Entrevistado e entrevistador no mesmo quadro.
- Entrevistador em *close*.
- Entrevistado em *close*.

O evento "entrevista" pode ser modelado de várias maneiras utilizando as diferentes opções de enquadramento. Quando só o primeiro tipo de enquadramento existe é mais fácil modelar o evento. Mas quando o segundo e o terceiro tipo de enquadramento estão presentes, a atribuição é mais difícil, pois é não possível determinar o relacionamento espacial entre os objeto de interesse e a ordem em que eles aparecem pode ser aleatória. A entrevista pode ocorrer com o enquadramento nos dois objetos de interesse. Desta forma o evento "entrevista" poderia ser descrito simplesmente pela ação semântica "Entrevistado permanece ao lado do entrevistador". No entanto, no momento que o entrevistador faz uma pergunta, pode existir uma mudança de enquadramento colocando o entrevistador em close. O mesmo pode ocorrer quando o entrevistado responde a pergunta. Um possível grafo que descreva o evento entrevista pode ser visto na Figura 34.



AS_r → Entrevistado e entrevistador lado a lado.

AS_s → Entrevistador em close.

AS_t → Entrevistado em close.

Figura 34 – Grafo que descreve o evento " Entrevista".

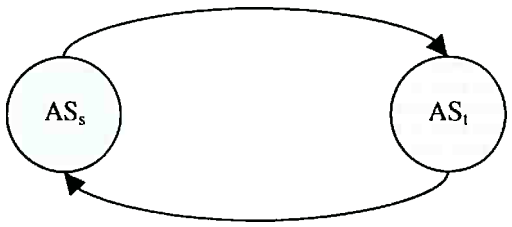
O grafo descreve um trecho de uma entrevista que começa com os dois objetos no mesmo quadro, passa pelo enquadramento do entrevistador (pergunta) e em seguida pelo enquadramento do entrevistado (resposta). O número de perguntas e respostas pode variar até o instante em que a entrevista termina com o enquadramento dos dois objetos. Este grafo restringe uma série de entrevistas que não seguem esta estrutura. Um exemplo que o grafo da Figura 34 atende pode ser visto na Figura 35.

Além de considerar uma ação semântica que interprete o objeto em close (por exemplo, "entrevistado está contido no quadro") e que, portanto não representa nenhuma relação espacial, é preciso que o processamento de imagem forneça atributos dos objetos de vídeo que permitam a distinção de quem é o entrevistado e de quem é entrevistador.

Figura 35 – Exemplo do evento "entrevista".

No exemplo existem dois ciclos onde o entrevistado ou o entrevistador estão em close (quadros Q1 e Q6). Se a ação descrita no primeiro quadro não ocorrer, o evento não será identificado com o grafo da Figura 34.

Para contornar o problema da restrição, poder-se-ia excluir o primeiro nó do grafo e o evento "entrevista" seria tratado simplesmente pelas ações de troca de enquadramentos. O grafo ficaria como o da Figura 36. O nó cinza indica qual o ocorre primeiro.



AS_s → Entrevistador em close.

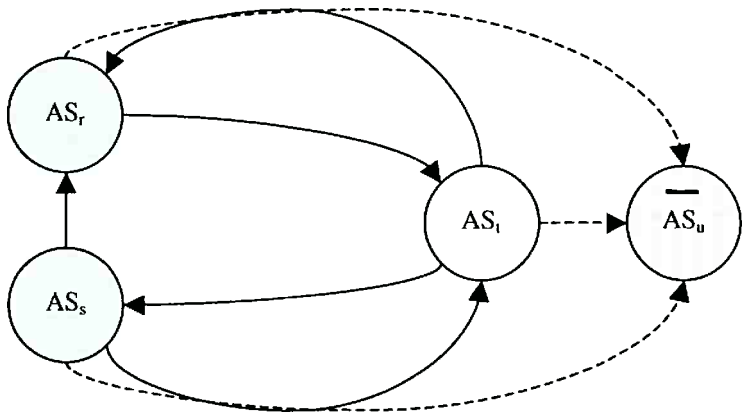
AS_t → Entrevistado em close.

Figura 36 – Grafo do evento "entrevista" modificado.

Com a alteração, o grafo modela uma conversa qualquer como, por exemplo, os diálogos de um filme. Para se obter uma busca mais precisa, o usuário teria que usar os atributos, local semântico e data semântica como pode ser visto na solicitação: "Retorne os trechos de vídeo onde ocorre o evento entrevista no programa do Jô".

Com a definição do local semântico somente trechos do "programa do Jô" seriam selecionados.

Outra representação do evento entrevista é visto na Figura 37.



AS_r → Entrevistado e entrevistador lado a lado.

AS_s → Entrevistador em close.

AS_t → Entrevistado em close.

AS_u → Objeto "entrevistado" está presente.

Figura 37 – Grafo do evento "entrevista".

O grafo possui múltiplos nós de inícios (AS_r e AS_s) e a interação entre entrevistado e entrevistador pode ocorrer várias vezes. A condição de término é a ausência do objeto entrevistado ($\overline{AS_u}$). A ação semântica AS_u representa uma condição de existência do objeto "entrevistado". A negação de AS_u é válida quando o objeto não é encontrado mais ao longo do vídeo e portanto, a sua entrevista terminou. No entanto se o objeto entrevistado aparecer novamente, o evento é instanciado novamente. Isto possibilita a composição do evento em partes. Isto também atende cortes de edição, problemas de sinal de transmissão ou simplesmente falta de seqüência gravada. É importante ressaltar que embora a ação AS_u faça parte do grafo que descreve o evento, ela não faz aparte do evento. Portanto, os trechos de vídeo retornados pela busca pelo evento entrevista, não contém as ação AS_u. O arco tracejado indica que a ação apontada por ele é necessária para o evento ocorrer (neste caso como condição de término), mas não faz parte do evento. Se os arco tracejados forem substituído por arcos contínuos, AS_u passa a fazer parte do evento.

5. CONCLUSÃO

O desenvolvimento de ferramentas para a extração de vídeos e segmento de vídeo é uma tendência irreversível. Grande sites de busca estão preparando ferramenta que permitam ao usuário pesquisar vídeos por palavras chaves. A indexação textual ainda traz uma série de problemas como, por exemplo, subjetividade e tempo despendido percorrendo o vídeo.

O modelo proposto eleva o problema de armazenamento, indexação e consulta de bancos de dados multimídia a um nível de abstração maior. Além de trabalhar com objetos de vídeo, o modelo se propõe a extrair conhecimento do modo como estes objetos se relacionam. Com o auxílio de um vocabulário já utilizado na literatura, as relações espaciais, temporais, espaço temporais e de existência podem ser usadas. O modo de descrição de um evento é baseado na atribuição de predicados aos trechos de vídeo. Esta abordagem faz com que modelo se afaste da programação procedural e se aproxime da programação declarativa utilizando lógica.

Com a inclusão de uma base de conhecimento onde as regras dos eventos são armazenadas é possível reduzir o tempo gasto com indexação dos vídeos. Embora uma parte do processo de indexação ainda seja de responsabilidade de um modelador (como é o caso da data e local do evento) a maior parte de identificação do evento pode ser feita automaticamente a partir da definição de regras. Um evento pode possuir múltiplas regras o que permite aprimorar o evento com o passar do tempo. Como a arquitetura dos eventos não é fixa, o modelo permite a sobreposição de eventos. O modelo foi desenvolvido com a intenção de não ser dependente do processamento de imagem podendo ser integrado com diferentes níveis de processamento de imagem. Desta forma, independente de como os trabalhos evoluírem nesta área o modelo poderá ser implementado.

Na análise de exemplos de casos com, por exemplo, eventos esportivos, o modelo consegue expressar com precisão os eventos. Isto porque eventos esportivos podem possuir fronteiras de início e fim muito bem definidas. Ou a bola entra no gol, ou a bola cruza a linha ou a bola passa pela cesta. Sempre podem ser descritos

através do comportamento das relações espaços-temporais. No caso da ausência de uma cena ou falha técnica, a identificação do evento fica comprometida.

O exemplo da entrevista foi escolhido para demonstrar que o modelo também consegue descrever eventos onde a relação espacial entre os objetos não define as fronteiras entre as ações. Neste caso foi preciso abstrair ainda mais para se determinar quais eram realmente as ações semânticas que compunham o evento entrevista e suas fronteiras.

No exemplo do furto, um ponto importante a ser destacado é o fato do predicado "entra" ser representado de forma diferente do mesmo predicado visto no exemplo "gol". A bola "entra" no gol ou a pessoa "entra" na sala indicam a mesma ação, porém como as câmeras estão posicionadas em locais diferentes. No evento gol é possível atribuir ao predicado "entra" a descrição definida nas relações espaciais como visto na Tabela 4. Mas no exemplo de furto, a câmera está posiciona dentro da sala e então a ação "entrar" limita-se ao aparecimento ou não de uma pessoa.

A continuidade do trabalho poderá ocorrer com a inclusão do áudio no modelo. Com isso a identificação de eventos pode ganhar em qualidade e rapidez uma vez que uma simples fala pode ser o suficiente para identificar um evento. Se um banco de vídeos por percorrido para a identificação do evento "casamento" a identificação de um trecho de áudio com os dizeres "Eu vos declaro marido e mulher" indica a possibilidade da ocorrência do evento casamento. A identificação de uma noiva e um noivo pode servir como prova da ocorrência do evento.

Outra possibilidade seria a implementação do sistema integrando o modelo com as mais variadas técnicas de processamento de imagem e áudio disponíveis.

ANEXO 1 - Tabela de transitividade para relações temporais.

A relação de igualdade "=" foi omitida da tabela.

B r2 C / A r1 B	<	>	d	di	o	oi	m	mi	s	si	f	fi
"before" <	<	no info	< o m d s	<	<	< o m d s	<	< o m d s	<	<	< o m d s	<
"after" >	no info	>	> oi mi d f	>	> oi mi d f	>	> oi mi d f	>	> oi mi d f	>	>	>
"during" d	<	>	d	no info	< o m d s	> oi mi d f	<	>	d	> oi mi d f	d	< o m d s
"contains" di	< o m di fi	> oi di mi si	o oi dur con =	di	o di fi	oi di si	o di fi	oi di si	di fi o	di	di si oi	di
"overlaps" o	<	> oi di mi si	o d s	< o m di fi	< o m	o oi dur con =	<	oi di si	o	di fi o	d s o	< o m
"over-lapped-by" oi	< o m di fi	>	oi d f	> oi mi di si	o oi dur con =	> oi mi	o di fi	>	oi d f	oi > mi	oi	oi di si
"meets" m	<	> oi mi di si	o d s	<	<	o d s	<	f fi =	m	m	d s o	<
"met-by" mi	< o m di fi	>	oi d f	>	oi d f	>	s si =	>	d f oi	>	mi	mi
"starts" s	<	>	d	< o m di fi	< o m	oi d f	<	mi	s	s si =	d	< m o
"started by" si	< o m di fi	>	oi d f	di	o di fi	oi	o di fi	mi	s si =	si	oi	di
"finishes" f	<	>	d	> oi mi di si	o d s	> oi mi	m	>	d	> oi mi	f	f fi =
"finished-by" fi	<	> oi mi di si	o d s	di	o	oi di si	m	si oi di	o	di	f fi =	fi

BIBLIOGRAFIA

1. ALLEN J. F., Maintaining Knowledge about Temporal Intervals, Communication. ACM, vol. 26, p. 832-843, Nov 1983.

2. BABAGUCHI, *N.;* KAWAI, Y.; Event Based Indexing of Broadcasted Sports Video by Intermodal Collaboration, IEEE Transactions on Multimedia, vol. 4, no. 1, Mar. 2002

3. BENITEZ, A.B.; RISING, H.; JORGENSEN, C. ; LEONARDI, R.; BUGATTI, A.; HASIDA, K.; MEHROTRA, R.; MURAT TEKALP, A.; EKIN, A. WALKER, T., Semantics of multimedia in MPEG-7, International Conference on Image Processing, vol.1 p. 137-140, 2002.

4. CHEN L.; ÖZSU M. T., Modeling of Video Objects in a Video Databases, IEEE International Conference on Multimedia and Expo, p. 171-175, 2002.

5. CHEN L.; ÖZSU M. T.; ORIA V., Modeling video Data for Content Based Queries: Extending the DISIMA Image Data Model, 1st. International Workshop on Computer Vision Meets Databases, Paper session, p.19-26, 2004.

6. CIOCCA, G.; SCHETTINI, R., Dynamic key-frame extraction for video summarization, Proc. Internet imaging VI, Vol. SPIE 5670, p. 137-142, 2005.

7. DÖNDERLER, M. E.; ŞAYKOL, E.; ULUSOY, Ö; GÜDÜKBAY, U., BilVideo: A Video Database Management System, IEEE Multimedia, Vol. 10, No. 1, p. 66-70, Mar 2003.

8. DURAK, N., Semantic Video Modeling and Retrieval with Visual, Auditory, Textual Sources, 2004, 95p. Dissertação (Mestrado), The Graduate School Of Natural And Applied Sciences Of Middle East Technical University, Set 2004.

9. EGENHOFER, M.; FRANZOSA, R., Point-set topological spatial relation, 1st. J. of Geographical Information Systems, vol.5 (2), p.161-174, 1991.

10. EKIN A.; TEKALP A. T.; MEHROTRA R., Integrated Semantic-Syntactic Video Modeling for Search and Browsing, IEEE Transaction on Multimedia vol. 6, no. 6, p. 839-851, Dez. 2004.

11. ERWIG, M.; SCHNEIDER, M., Spatio-Temporal Predicates, IEEE Transactions on Knowledge and Data Engineering, vol14 , p. 881-901, 2002.

12. ERWIG, M.; SCHNEIDER, R.; Developments in spatio-temporal query languages. In Proceeding of DEXA Workshop on Spatio-Temporal Data Models and Languages, p 441-449, 1999.

13. FADO, D.; LYONS, B.; PENKER, M.; ERIKSSON, H.R. UML 2.0 Tookit, .ed. Indianapolis: Wiley Publishing, Inc, 2004.

14. HAERING, N; QIAN, R.; SEZAN, M., A Semantic Event-Detection Approach and its Application to Detecting Hunts in Wildlife Video, IEEE Transactions on Circuits and Systems for Video Technology, Vol.10 n.6, p. 857-868, Set 2000.

15. KÖPRÜLÜ M.; ÇIÇEKLI, N. K.; YAZICI, A., Spatio-Temporal Querying in Video Database, Information and Computer Science: An International Journal, 160(1-4), p. 131-152, Mar 2004.

16. LI, J. Z.; GORALWALLA, I. A.; ÖZSU, M.T.; SZAFRON, D., Modeling Video Temporal Relationship in an Object Database Management System, International Symposium on Electronic Imaging: Multimedia Computing and Networking, p. 80-91. Fev 1997.

17. LI (A), J. Z.; ÖZSU, M. T.; SZAFRON D., Spatial Reasoning Rules in Multimedia Management System, International Conference on Multimedia Modeling, Toulouse, France, p. 119-133, Nov 1996.

18. LI (B), J. Z.; ÖZSU, M. T.; SZAFRON, D., Modeling of Video Spatial Relationship in an object Database Management System, IEEE 1st. Workshop on Multimedia. Database Management System, p. 124-132, 1996.

19. MARQUES, O.; FURHT, B. Introduction to Video Databases, Handbook of Video Databases, Editors-in-Chief: Borko Furht and Oge Marques, CRC Press, Cap 1, 2004.

20. NARASIMHA, R.; SAVAKIS, A.; Key Frame Extraction using MPEG-7 Motion Descriptors, Asilomar Conf., Nov. 2003.

21. OOMOTO, E; TANAKA, K, OVID: Design and Implementation of a Video-Object Database System", IEEE Transaction on Knowledge and Data Engineering, vol. 5, no. 4, p. 629-643, Ago. 1993.

22. PALAZZO, L.; Introdução a Programação Prolog; Editora da Universidade Católica de Pelotas, 1997.

23. PAPADIAS, D.; SELLIS, T., Qualitative Representation of Spatial Knowledge in Two-Dimensional Space, VLDB Journal 3, p. 479-516, 1994.

24. PETKOVIC, M.; JONKER, W., Content-based video retrieval by integrating spatio-temporal and stochastic recognition of events. IEEE International. Workshop on Detection and Recognition of Events in Video, p. 75-82, Jul 2001.

25. Radio Televisione Italiana. Disponibiliza o vídeo do gol que contém os quadros utilizados no estudo de caso evento "gol". Disponível em: < http://www2.raisport.rai.it/news/rubriche/coppe978/200001/14/387f3f4707207/5mancini.mpg>. Acessado em 10 de Junho de 2005.

26. SALEMBIER P.; SMITH J. R, Mpeg-7 multimedia description schemes, IEEE transactions on circuits and systems for video technology, vol. 11, no. 6, p. 748-759, Jun 2001.

27. SCOTT, K., Fast Track UML 2.0., Berkeley: Apress, 2004.

28. SILBERSCHATZ, A; KORTH, H; SUDARSHAN, S.; Database System Concepts; McGraw Hill, 1996.

29. SINTE – Sindicado dos Terapeutas. Disponibiliza o vídeo com da entrevista utilizado no estudo de caso do evento "entrevista". Disponível em: < http://www.sinte.com.br/videos/JoSoares11eMeiaQuiropatia.asf>. Acessado em 10 de Junho de 2005.

30. Telemetrika. Disponibiliza o vídeo do furto que contem os quadros utilizados no estudo de caso do evento "furto". Disponível em: < http://www.telemetrika.com/VideoSamples.html>. Acessado em 10 de Junho de 2005.

31. WAHL, T.; ROTHERMEL, K., Representing Time in Multimedia-Systems, IEEE International Conference on Multimedia Computing and Systems, p. 538-543, 1994.