

## 5

### Descritores para posicionamento de câmera

Neste capítulo focamos no problema de otimização de posicionamento de câmera em cenas tridimensionais. Conforme descrito anteriormente, as Galerias Inteligentes extraem uma coleção de descritores dos parâmetros selecionados. Estes descritores devem representar, usando números reais, características do resultado obtido com um determinado parâmetro. Portanto, é fundamental a escolha de uma coleção de descritores que represente as principais características dos resultados do parâmetro, de modo que a máquina de aprendizagem seja capaz de interpretar e identificar, através de números, as preferências do usuário.

Para otimizar posicionamento de câmera, utilizamos descritores já existentes na literatura, com algumas variações, e desenvolvemos outros para expandir a capacidade de interpretação da máquina para este problema. Utilizamos também descritores para aplicações específicas, como um descritor que mede a velocidade visível em simulação de fluidos. Os descritores selecionados apresentaram resultados satisfatórios em nossos experimentos (ver capítulo 6). Porém, outros descritores podem ser tão ou mais eficientes, em particular se forem consideradas informação de cor, textura e outras características não geométricas. Procuramos apenas colocar o máximo de descritores que foi possível implementar. Esses descritores podem ser classificados de acordo com o domínio em que avaliam os parâmetros:

- Descritores do espaço do objeto: avaliam propriedades (visíveis) dos objetos da cena, como curvatura média e componente conexa;
- Descritores do espaço da imagem: extraem características da imagem renderizada, como a área projetada do objeto na imagem, seu alinhamento em relação aos eixos da imagem e a complexidade da silhueta do objeto.

A primeira seção trata da computação eficiente desses descritores, com o objetivo de possibilitar o uso das Galerias Inteligentes em tempo real ou quasi. Em seguida, descrevemos detalhadamente os descritores utilizados neste

trabalho. Adiantamos que os descritores que se mostraram influentes em boa parte dos experimentos foram os relacionados a propriedades de curvatura e saliência, e o descritor de área projetada.

## 5.1

### Extração eficiente dos descritores

Dado um parâmetro de câmera  $\omega_i$ , cada descritor  $\varphi_j$  do espaço do objeto realiza um processamento sobre os pontos visíveis da visão resultante  $\pi(\omega_i)$ , ou seja, na imagem renderizada. Cada descritor está diretamente associado a alguma propriedade dos pontos visíveis, como curvatura média, componente conexa ou *patch*. Essas propriedades são calculadas, normalizadas e quantizadas usando uma determinada quantidade de bits durante a fase de pré-processamento.

Para cada propriedade calculada nos pontos do objeto, geramos e equalizamos seu histograma, para melhorar a distribuição dos valores sobre os objetos. Seja  $h_x(a) = \sum_{k=0}^a p_x(k)$  a função de distribuição acumulada da propriedade  $x$  com probabilidades  $p_x$ , de acordo com seu histograma original. Para equalizá-lo, queremos encontrar a transformação  $T(x) = y$ , tal que a nova função de distribuição acumulada  $h_y$  cresça linearmente, ou seja,  $h_y(a) \propto a$ . É fácil ver que  $T(x) = h_x(x)$  respeita nossa condição.

Para acelerar a extração das propriedades dos pontos visíveis, quantizamos estas propriedades dos vértices do objeto e em seguida concatenamos as propriedades quantizadas em cores de 32 bits no formato RGBA. Finalmente, renderizamos uma imagem com as cores codificadas.

A imagem codificada é extraída do *buffer* de cores e dada como entrada para cada descritor, seja ele de atuação no espaço do objeto ou da imagem. Cada descritor saberá decodificar propriedades de interesse nos pixels da imagem. Note que cada pixel não vazio desta imagem corresponde a um ponto visível. Essa estratégia reduz o tempo de processamento dos descritores por imagem, possibilitando a interação em tempo real com as galerias (ver capítulo 6).

A figura 5.2 (esquerda) mostra um exemplo de uma imagem codificada apenas com a curvatura gaussiana, onde os pixels mais escuros representam pontos de alta curvatura. Em nosso caso, foram necessárias 2 imagens codificadas por visão para representar todas as propriedades utilizada com uma quantização adequada. A tabela 5.1 exibe os principais descritores, suas propriedades correspondentes e a quantidade de bits utilizada na codificação.

propriedade	descriptor	# bits
curvatura média	$\Sigma$ baixas curvaturas visíveis	6
	$\Sigma$ médias curvaturas visíveis	
	$\Sigma$ altas curvaturas visíveis	
saliência	$\Sigma$ baixas saliências visíveis	6
	$\Sigma$ médias saliências visíveis	
	$\Sigma$ altas saliências visíveis	
patches	$\Sigma$ patches visíveis	10
	$\Sigma$ área 2D visível / área 3D	
componente conexa	da componente conexa	10
	$\Sigma$ área 2D visível $\times$ área 3D da componente conexa	

Tabela 5.1: Propriedades e descritores: cada descriptor está associado a uma propriedade dos vértices da cena. Estas propriedades dos vértices são quantizadas usando uma determinada quantidade de bits e codificadas em cores de 32 bits no formato RGBA.

## 5.2

### Descritores do espaço do objeto

Nesta classe de descritores, medimos a visibilidade das propriedades geométricas ou inerentemente tridimensionais dos pontos visíveis, tais como curvatura média, saliência, componentes conexas e *patches*. Desenvolvemos também dois descritores para aplicações específicas: velocidade em simulação de fluidos e visibilidade de grupos em uma animação de batalha.

#### 5.2.1

##### Descritores geométricos

Boas visões de uma cena tridimensional em geral estão correlacionadas com a quantidade de informação geométrica relevante visível (46, 34, 23). Uma abordagem para medir a quantidade de informação geométrica relevante em uma superfície é baseada na presença de feições (*features*), ou seja, pontos de alta curvatura, como pontos que fazem parte de arestas e quinas. Porém, nem sempre regiões ricas em pontos de alta curvatura são geometricamente importantes, no sentido de representarem partes salientes da geometria do objeto, de acordo com nossa percepção. Um exemplo clássico é exibido na figura 5.1, onde a região do cabelo do David apresenta alta curvatura, porém não se trata de uma região rica em saliências. Por esta razão, utilizamos neste trabalho dois descritores que se complementam, para medir propriedades geométricas de pontos visíveis.

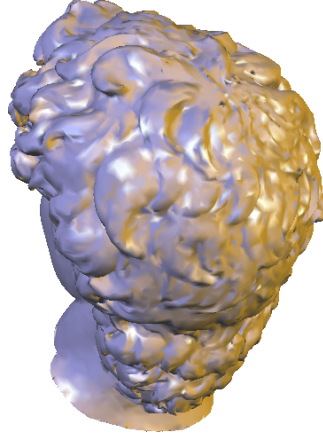


Figura 5.1: Visão traseira da cabeça do David: rica em alta curvatura mas pobre em saliências.

**Curvatura** O primeiro descritor é baseado na magnitude da curvatura média  $|\kappa_H|$  dos pontos visíveis. Durante a fase de pré-processamento, usamos o trabalho de Meyer *et al.* (28) para calcular em cada ponto  $x_v$ , o operador de Laplace-Beltrami discreto  $\mathcal{L}$  com o peso das cotangentes, dado por:

$$\mathcal{L}(q_v) = \frac{1}{2A_{Mixed}(q_v)} \sum_{q_w \in N_1(q_v)} (\cot \beta_{vw} + \cot \gamma_{vw})(q_v - q_w), \quad (5-1)$$

onde  $A_{Mixed}(q_v)$  é a região de Voronoi do ponto  $q_v$  e  $N_1(q_v)$  são os vértices de sua primeira vizinhança estrelada. Daí, a magnitude da curvatura média é dada por  $|\kappa_H(q_v)| = \frac{1}{2} \|\mathcal{L}(q_v)\|$ . Após o cálculo das curvaturas em todos os pontos, seu histograma é calculado e equalizado.

**Saliência** O segundo descritor de propriedades geométricas usa uma medida de saliência derivada da curvatura média, apresentada por Lee *et al.* (23). Essa medida de saliência combina suavizações gaussianas da curvatura média em várias escalas  $\iota$ , dadas pela equação abaixo:

$$\mathcal{S}(q_v, \iota) = \frac{\sum_{x_w \in N(q_v, 2\iota)} \kappa_H(q_w) \exp\left(\frac{-\|q_w - q_v\|^2}{2\iota^2}\right)}{\sum_{q_w \in N(q_v, 2\iota)} \exp\left(\frac{-\|q_w - q_v\|^2}{2\iota^2}\right)}. \quad (5-2)$$

Após o cálculo das saliências em cada ponto, seu histograma é calculado e seus valores equalizados. A figura 5.2 (direita) exhibe um exemplo de imagem codificada com as saliências, sendo os pontos escuros seus pontos mais salientes.

Para extrair mais características das visões, dividimos essas duas propriedades geométricas em três intervalos: baixos valores  $[0, \frac{1}{3})$ , médios valores  $[\frac{1}{3}, \frac{2}{3})$  e altos valores  $[\frac{2}{3}, 1]$ . Finalmente, cada descritor soma a quantidade de pixels



Figura 5.2: Descritores geométricos do espaço do objeto: curvatura gaussiana (esquerda) e saliência (direita).

na imagem contendo valores em seu respectivo intervalo, como na Tabela 5.1. Intuitivamente, a máquina poderá aprender se uma dada visão contém grandes (ou poucas) regiões de alta curvatura, média curvatura, baixa curvatura, alta saliência, média saliência e baixa saliência.

Analogamente, tratamos a velocidade dos vértices da malha de um fluido, em um cenário de simulação de fluidos. Neste caso, a velocidade já é um dado de entrada associado a cada vértice, como codificado em cores na figura 2.3.

## 5.2.2

### Descritores de visibilidade de partes

Uma outra linha de descritores do espaço do objeto trata de medir a visibilidade de partes da cena tridimensional. Uma característica importante de uma visão é a área 3D visível. Porém, em algumas aplicações pode ser mais importante manter a maior quantidade de componentes conexas visíveis, independente da área total visível. Em outro cenário, pode ser mais importante ter as componentes conexas de maior área visíveis do que as de menor área, em especial quando a cena tiver grande potencial para oclusões, como ilustrado na figura 5.3.

Pensando nesses problemas, desenvolvemos três descritores, que usam como propriedades um identificador da sua componente conexa, calculado no processo de geração da malha (22), e um identificador de seu *patch*, o qual é gerado na fase de pré-processamento, e descrevemos a seguir.

Seguindo uma estratégia gulosa, inicializamos o *patch* de cada triângulo como ele mesmo. Unimos os patches de menor área usando operações *union-find* (44), de modo a manter os patches com aproximadamente a mesma área, até atingir um determinado limiar. No final do processo, cada triângulo estará associado a um identificador de seu *patch*.

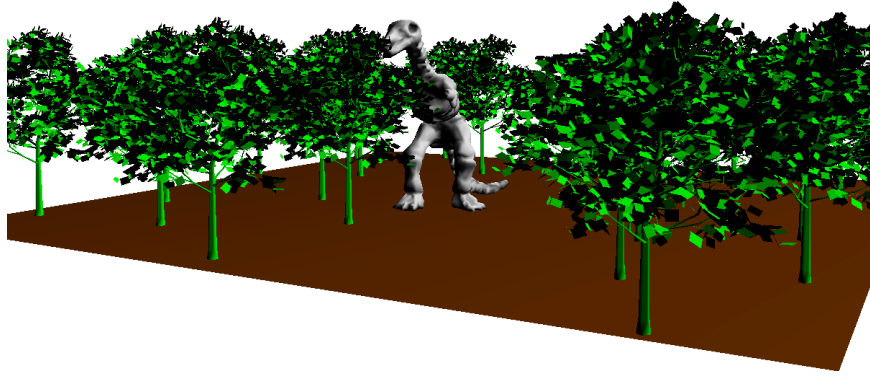


Figura 5.3: Cena rica em oclusão: o dinossauro se esconde atrás das árvores.

Uma outra propriedade que poderia ser utilizada identificaria cada vértice com um segmento da malha. Estes segmentos poderiam ser obtidos usando métodos convencionais de segmentação de malhas (38).

**Área 3D visível** O primeiro descritor aproxima a área 3D visível medindo a quantidade de diferentes *patches* visíveis na imagem. Como os *patches* têm aproximadamente a mesma área, contar a quantidade de *patches* visíveis nos dá uma boa aproximação da área 3D visível. É importante notar que se os *patches* gerados forem muito pequenos, corre-se o risco de que a projeção de um *patch* na imagem seja menor que a área de um pixel, e desse modo não seria contabilizado. Além disso, uma grande quantidade de *patches* aumenta a quantidade de bits quantizados na imagem para representá-lo. Por outro lado, *patches* grandes aumentam o erro no cálculo da área 3D causado por *patches* parcialmente visíveis. A figura 5.4 exibe uma imagem codificada com *patches*.

**Visibilidade de componentes conexas** O segundo descritor desta linha tem o objetivo de medir a quantidade de componentes visíveis, e atua somando a área projetada visível das componentes, normalizada por sua área 3D, como na Tabela 5.1.



Figura 5.4: Descritores de visibilidade de partes do espaço do objeto: *patches* (esquerda) e grupos (direita)

**Área relativa** Finalmente, para enfatizar grandes componentes, desenvolvemos um descritor que soma a área projetada visível das componentes multiplicada por um peso dado por sua área 3D.

Em uma aplicação específica de animação, usamos como identificador das partes o grupo ao qual vários guerreiros pertencem (figura 5.4)

### 5.3

#### Descritores do espaço da imagem

Assim como os descritores do espaço do objeto, os descritores do espaço da imagem usam propriedades dos pontos projetados na imagem. Em particular, serão utilizadas as curvaturas médias e componentes conexas.

**Área projetada** O primeiro descritor desta linha mede a área projetada no espaço da imagem, somando os pixels com componentes conexas não-nulas.

**Silhueta** Um outro tipo de característica de uma visão que está relacionada com a percepção do objeto tridimensional pelo observador é a complexidade de sua silhueta. Em geral, silhuetas mais complexas fornecem mais informação ao observador, como ilustrado na figura 5.5, onde é possível identificar claramente a silhueta de um coelho. Definimos o descritor de complexidade da silhueta como a integral do valor absoluto de sua curvatura. Usamos um método simples de continuação para extrair sua silhueta, e somamos os valores absolutos de suas curvaturas em todos os seus pontos  $pix_t$ , com peso dado pelo comprimento do segmento  $\overline{pix_t pix_{t+1}}$ . Usamos o método de Lewiner *et al.* (24) para calcular as curvaturas.

Duas informações importantes de uma visão não foram consideradas até então: a posição dos objetos na imagem, em particular de feições do objeto; e a orientação da imagem.

**Centralidade de feições** Para obter a primeira informação, criamos um descritor que representa a distribuição dos pontos, dando ênfase aos pontos de alta curvatura média. Este descritor calcula a distância média da posição projetada dos pontos visíveis ao centro da imagem, ponderada por sua curvatura média, ou seja:

$$\varphi_j(\omega_i) = \frac{1}{|\mathcal{V}_i|} \sum_{pix \in \mathcal{V}_i} |\kappa_H(q(pix))| \|pix - center\|,$$

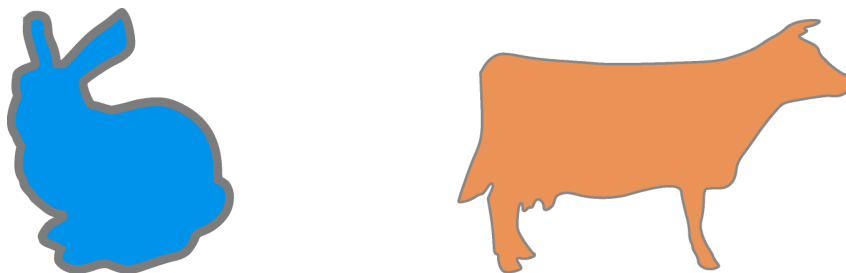


Figura 5.5: Descritores do espaço da imagem: coelho com silhueta complexa (esquerda) e vaca bem orientada em relação aos eixos da imagem (direita).

onde  $pix$  é um pixel não nulo da imagem  $\pi(\omega)$  representando a projeção de um ponto  $q(pix)$  do objeto,  $center$  é o centro da imagem e  $\mathcal{V}_i$  o conjunto de pixels não nulos da imagem  $\pi(\omega_i)$ . Este descritor retorna valores baixos se os *features* do objeto estão centralizados na origem, e valores mais altos se estão na borda da imagem.

**Orientação das componentes conexas** Finalmente, realizamos regressões lineares sobre os pixels correspondentes a cada componente conexo visível na imagem. A inclinação destas retas nos diz se os componentes estão alinhados com os eixos da imagem. Para destacar grandes componentes conexas, este descritor é definido como a média das inclinações destas retas ponderada pelo tamanho de seus respectivos componentes. Na figura 5.5 exibimos um exemplo de uma vaca bem alinhada com os eixos da imagem.