

7 Sistema de Predição Adaptativo

No capítulo anterior foi apresentada a arquitetura do sistema de gerenciamento de memória. A estrutura baseada em módulos permite que o sistema seja utilizado em vários tipos de aplicação. A implementação dos módulos que compõem o sistema de paginação é que o torna mais adequado para um determinado grupo de aplicações.

Este capítulo apresenta um sistema de paginação voltado para aplicações de visualização em tempo-real. O sistema de paginação foi desenvolvido para lidar apenas com textura. Esse sistema é adequado para aplicações que necessitam visualizar texturas de altíssimas resoluções em tempo-real. Como por exemplo, aplicações para visualização de imagens de satélites e panoramas virtuais.

O sistema de predição é implementado dentro do sistema de carregamento de páginas. Esse sistema é responsável por carregar as páginas que provavelmente serão necessárias num futuro próximo. Certamente a melhor forma de estudar esse problema é através de um modelo de probabilidade. Em geral, esses modelos são descritos a partir de modelos matemáticos sofisticados. Devido ao pouco tempo de processamento destinado ao cálculo de predição, é necessário criar um modelo mais simples que obtenha resultados próximos aos obtidos pelos modelos probabilísticos.

As aplicações de visualização acessam os dados de acordo com os movimentos realizados pela câmera virtual. Por outro lado, existe uma quantidade máxima de dados que pode estar disponível no estágio mais alto de armazenamento num instante de tempo. A técnica de predição apresentada neste trabalho explora essas informações (coerência temporal) e a estrutura em multi-resolução (coerência espacial) para elaborar um modelo matemático simples para estimar os dados que serão necessários num futuro próximo.

O sistema de liberação de páginas é tão importante quanto o sistema de carregamento. Decisões erradas podem prejudicar o desempenho da aplicação e do sistema. Isto ocorre porque páginas que são retiradas

indevidamente, serão carregadas novamente pelo sistema. Ou seja, o tempo que poderia ser gasto carregando uma informação a mais, é gasto carregando a página que acabou de ser retirada. Para evitar esse problema, é proposto um algoritmo de liberação que se baseia nos parâmetros da câmera para decidir quais os dados que não serão utilizados num futuro próximo.

O sistema de tratamento de ausência de páginas foi desenvolvido com o objetivo de evitar que a aplicação seja interrompida no momento em que uma página de dados não carregada seja acessada. Esse problema será resolvido explorando a representação em multi-resolução do objeto gráfico.

Além dos sistemas, é necessário criar regras para as operações de paginação. Essas regras têm como objetivo manter a coerência entre as operações de carregamento, liberação e tratamento de ausência de páginas.

7.1

Sistema de Carregamento de Páginas

O sistema de carregamento foi projetado e implementado para lidar com aplicações de visualização de objetos gráficos. Desta forma, o sistema baseia-se nas informações vindas do sistema de visualização e do objeto gráfico para escolher quais páginas de textura devem ser carregadas. A figura 7.1 mostra a arquitetura do sistema.

Esse sistema é basicamente formado por três componentes:

- **Sistema de predição:** Tem a função de monitorar os parâmetros da câmera e, baseado nesta informação, determinar a região do objeto gráfico e o nível de resolução que provavelmente serão acessados pelo sistema de visualização num futuro próximo. As texturas dos ladrilhos dentro dessa região serão carregadas para o estágio mais alto de armazenamento.
- **Sistema de regras de carregamento:** É responsável por efetivamente carregar as páginas de textura escolhidas pelo sistema de predição. O carregamento de cada página será realizado levando em consideração um conjunto de regras. Esse sistema também realiza as operações de cancelamento.
- **Objeto Gráfico:** Tem a função de fornecer os ladrilhos de textura que estão dentro de uma região retangular. Essa região é determinada pelo sistema de predição. A decisão de deixar esta tarefa com o objeto gráfico foi com o objetivo de livrar o sistema de carregamento de lidar com problemas relacionados com visualização, geometria e

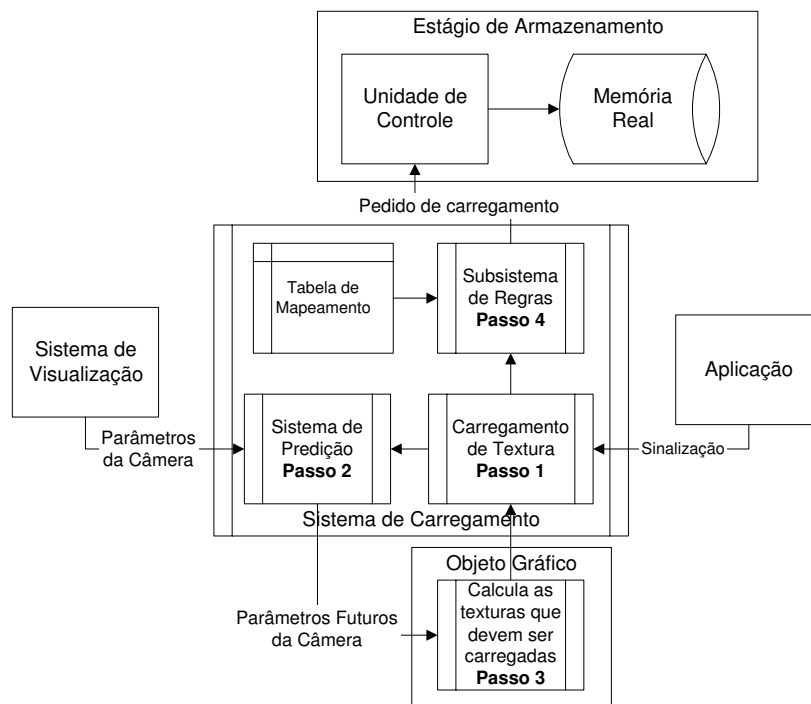


Figura 7.1: Arquitetura e funcionamento do sistema de carregamento de páginas para aplicações de visualização.

topologia de objetos gráficos. Isto torna o sistema de carregamento mais genérico, podendo lidar com qualquer tipo de visualização e objeto gráfico.

O sistema de carregamento também possui algumas estruturas internas como:

- Um vetor de carregamento. Esse vetor armazena os endereços virtuais das texturas que devem ser carregadas num futuro próximo.
- Um vetor de cancelamento. Esse vetor armazena os endereços virtuais das texturas que não devem ser mais carregadas.
- Um relógio para cronometrar o tempo de processamento do sistema de carregamento.

Ao criar um sistema de carregamento é necessário fornecer a estrutura `ObjectMemory`, um sistema de regras e o número máximo de quadros que o sistema tem que prever. O funcionamento do sistema é realizado a partir dos seguintes passos:

1. Inicia o cronômetro de tempo.

2. Chama o sistema de predição que atualiza os vetores de carregamento e cancelamento.
3. Varre o vetor de cancelamento e para cada endereço chama o sistema de regras para cancelar o pedido de carregamento.
4. Varre o vetor de carregamento e para cada endereço chama o sistema de regras para realizar o pedido de carregamento.

As operações de carregamento e cancelamento são realizadas até terminar o tempo de processamento dado para o sistema de gerenciamento de memória.

7.1.1 Sistema de Predição

Como já foi discutido, o sistema de predição tem a finalidade de determinar quais os ladrilhos de textura que serão acessados num futuro próximo. Os endereços virtuais desses ladrilhos são passados para o sistema de carregamento. Em seguida, o sistema de carregamento providenciará que os dados associados a esses endereços sejam carregados.

Nas aplicações de visualização os acessos aos dados estão relacionados com os movimentos da câmera virtual. Isto ocorre porque os dados que estão dentro do campo de visão da câmera são processados e convertidos em informação visual.

O sistema de predição recebe como parâmetros um ponteiro para câmera virtual, um ponteiro para estrutura `ObjectMemory` e a quantidade de quadros futuros que o sistema deve prever. O ponteiro para câmera é utilizado para monitorar as variações sofridas em seus parâmetros. O ponteiro para a estrutura `ObjectMemory` é utilizado para o sistema trocar informações com o objeto gráfico.

A operação de predição é realizada em três passos. O primeiro consiste em determinar a região do objeto gráfico que possui a maior probabilidade de ser acessada. Essa região é chamada de *área de carregamento*. O segundo é determinar o melhor nível de resolução que deve ser carregado. O terceiro consiste em determinar quais os pedidos de carregamento que devem ser cancelados. A região onde os pedidos são cancelados é chamada de *área de descarregamento*.

A área de carregamento é determinada pela união das janelas de visão calculadas nos quadros futuros, como é ilustrado na Figura 7.2. As janelas de visão são estimadas a partir dos cálculos de velocidade e aceleração dos parâmetros da câmera virtual. Supondo que o quadro atual seja i ,

o sistema de predição calcula os parâmetros da câmera para os quadros futuros $[i + 1, i + 2, \dots, i + n]$, onde n é o número de quadros que devem ser previstos. Em seguida, cada parâmetro futuro é passado para o objeto gráfico que retorna as coordenadas da janela de visão em relação ao sistema de coordenadas do seu suporte paramétrico. No Capítulo 5 é mostrado como converter os parâmetros da câmera em uma janela de visão para o caso das imagens de satélites e dos panoramas virtuais.

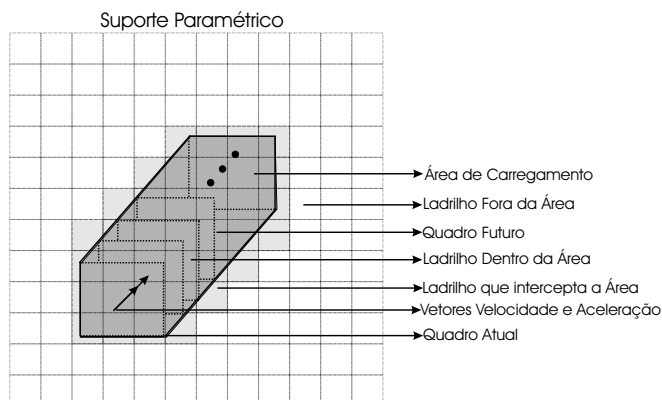


Figura 7.2: Os ladrilhos que estão dentro ou que interceptam a área de carregamento possuem uma probabilidade maior de serem requeridos pela aplicação de visualização e devem ser carregados.

O nível de resolução dentro da área de carregamento é escolhido considerando critérios da taxa de transmissão e de distância. O critério da taxa de transmissão é baseado no número de ladrilhos por quadro que podem ser carregados para o estágio mais alto de armazenamento. O critério de distância escolhe os ladrilhos de menor resolução nas regiões da área de carregamento que estão distantes da posição atual da câmera.

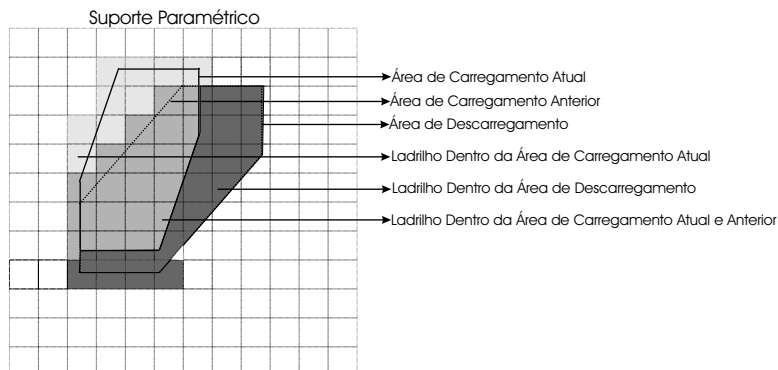


Figura 7.3: Os pedidos de carregamento dos endereços virtuais dos ladrilhos que estão dentro da área de descarregamento são cancelados.

A área de descarregamento é região determinada pela diferença entre área de carregamento calculada no quadro anterior e a área calculada no quadro atual, como mostra a Figura 7.3. Os pedidos de carregamento dos ladrilhos que estão dentro dessa área são cancelados.

Parâmetros Futuros da Câmera

Matematicamente, o problema de prever os valores dos parâmetros da câmera a partir da análise de valores assumidos no passado pode ser colocado da seguinte forma: Sejam uma seqüência uniforme de instantes de tempo $(t_0, t_1, t_2, \dots, t_n)$, onde $t_i \in \mathbb{R}^+$, uma seqüência de parâmetros $(p_0, p_1, p_2, \dots, p_n)$, onde $p_i \in \mathbb{R}^n$, e uma função de mapeamento f definida como $f(t_i) = p_i$, onde $0 \leq i \leq n$. Encontrar o valor da função f no instante t_{n+k} , onde $k > 0$.

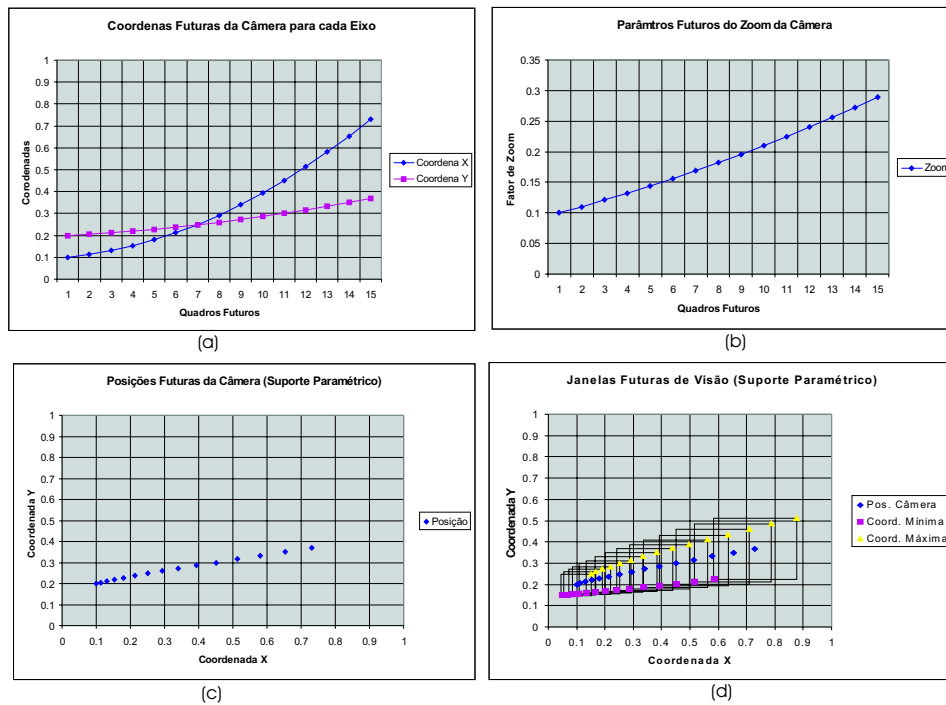


Figura 7.4: A predição dos parâmetros futuros da câmera para visualização de imagens de satélite;(a) As coordenadas futuras para cada grau de liberdade da câmera; (b) Os parâmetros futuros de zoom; (c) A predição das posições futuras da câmera dentro do suporte paramétrico; (d) As janelas de visão dentro da suporte paramétrico calculadas a partir dos parâmetros futuros da câmera.

Para calcular o valor correto da função f no instante $n + k$ é necessário impor restrições de como os parâmetros da câmera variam com o

tempo. Uma restrição comum é impor que a variação desses parâmetros esteja associada a algum tipo de fenômeno conhecido e que tenha um comportamento que seja bem determinado. Neste trabalho, a função f é descrita através das leis da mecânica. Assim, o sistema de predição utiliza as equações que descrevem o fenômeno de movimento do mundo físico. Os parâmetros futuros da câmera são calculados através dos seguintes passos:

1. Calcula-se as velocidades $v_{n-1} = \frac{p_{n-2}-p_{n-1}}{t_{n-2}-t_{n-1}}$ e $v_n = \frac{p_n-p_{n-1}}{t_{n-1}-t_n}$ e a aceleração $a_n = \frac{v_n-v_{n-1}}{t_n-t_{n-1}}$.
2. Em seguida, a função f é calculada como $f(t_{n+k}) = p_n + v_n t_{n+k} + \frac{a_n t_{n+k}^2}{2} = p_{n+k}$, onde $k > 0$.

Com o objetivo de simplificar os cálculos e torná-los independente do tempo com que os quadros são sintetizados, a velocidade é descrita por *unidades/quadro* e a aceleração é descrita por *unidades/quadro²*. No caso de visualização de panoramas a *unidade* é definida por ângulos em graus, logo tem-se que a velocidade é *graus/quadro* e a aceleração é *graus/quadro²*. Essa abordagem também elimina os problemas de imprecisão que existe na tomada de tempo de tarefas que levam muito pouco tempo para serem realizadas. Desta forma, o valor do parâmetro t representa o quadro que se deseja prever. Observe que $t = 0$ representa os parâmetros da câmera no quadro atual e que $t > 0$ representa os parâmetros da câmera nos quadros futuros.

O sistema de predição possui um vetor capaz de armazenar os parâmetros da câmera dos três últimos quadros. Com as informações desse vetor é possível calcular a função f descrita acima. Suponha que o vetor de parâmetros seja definido por $Buffer[0] = p_{n-2}$, $Buffer[1] = p_{n-1}$ e $Buffer[2] = p_n$. Agora, basta reescrever as equações descritas acima para notação vetorial. Para cada novo quadro sintetizado, o vetor $Buffer$ é atualizado com uma operação de deslocamento para a esquerda, isto é, $Buffer[0] = p_{n-1}$, $Buffer[1] = p_n$ e $Buffer[2] = p_{n+1}$, onde p_{n+1} representa os parâmetros da câmera nesse novo quadro. A Figura 7.4 mostra o processo de predição para o modelo de câmera virtual utilizado para visualizar imagens de satélite (Capítulo 5). A predição foi realizada para os primeiros 15 quadros. Os cálculos foram realizados considerando os seguintes parâmetros:

- O suporte paramétrico está definido no espaço $[0, 1] \times [0, 1]$.
- O vetor de velocidade inicial da câmera é $V_{eyes} = (0.01, 0.005)$ e o vetor de aceleração é $A_{eyes} = (0.005, 0.001)$.

- A velocidade inicial do fator de zoom é $V_{zoom} = 0.01$ e a aceleração é $A_{zoom} = 0.0005$.

Área de Carregamento

Como já foi discutido, a área de carregamento é determinada pela união das janelas de visualização calculadas a partir dos parâmetros futuros da câmera. O cálculo preciso dessa área pode gastar uma quantidade considerável do tempo de processamento destinado ao sistema de predição. Os fatores de influenciam nos cálculos são os seguintes:

- A rotina do objeto gráfico que converte um parâmetro da câmera em uma janela de visão é chamada várias vezes.
- Os parâmetros futuros da câmera são calculados a partir de uma equação de segundo grau. Isto implica que área de carregamento pode ser mais complexa do que o desejado. A Figura 7.5 mostra um exemplo de uma área com curvatura.
- O cálculo realizado para saber se um ladrilho está dentro área de carregamento se torna mais complexo.

Observe que os cálculos realizados nessa fase devem gastar o mínimo de tempo possível, porque é necessário deixar uma boa fatia do tempo para efetuar o carregamento dos dados. Desta forma, optou-se por simplificar os cálculos e gerar áreas de carregamento sem curvatura. A simplificação é capaz de produzir áreas de carregamento muito próximas das área reais em casos onde o ângulo entre o vetor de velocidade e o vetor de aceleração estão próximos de 0° ou 180° graus. Esses cálculos são ideais para as aplicações que utilizam dispositivos de entrada (mouse, joystick, etc.) para controlar os movimentos da câmera virtual.

A proposta de simplificação consiste em determinar a área de carregamento a partir de duas janelas de visão. Os cálculos são realizados levando em consideração três situações. No primeiro caso o vetor velocidade é nulo. A segunda situação ocorre quando o ângulo entre os vetores de velocidade e de aceleração é menor que noventa graus. Isto implica que a câmera está em movimento acelerado e a sua trajetória não vai mudar de direção. A terceira situação ocorre quando o ângulo entre os vetores de velocidade e de aceleração é maior que noventa graus. Neste caso, a câmera está em movimento desacelerado e existe a possibilidade da sua trajetória mudar de direção. Para todos os casos será considerado que o sistema fará

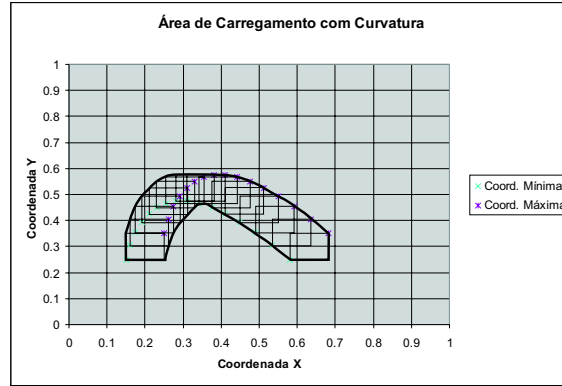


Figura 7.5: Área de carregamento com curvatura. Nesse caso o ângulo entre o vetor de velocidade e o vetor de aceleração é de 90° graus.

a predição para os N_q quadros futuros. Assim, o parâmetro de tempo t está dentro do intervalo $[0, N_q - 1]$.

Na primeira situação, onde o vetor de velocidade é nulo, as coordenadas da área de carregamento é igual as coordenadas da janela de visão calculada em $t = 0$.

Na segunda situação, a câmera em movimento acelerado, a área de carregamento é calculada a partir das janelas de visão obtidas em $t_i = 0$ e $t_f = N_q - 1$. Suponha que o retângulo envolvente da janela de visão em $t_i = 0$ tenha coordenadas $Min_i = (x_{mini}, y_{mini})$ e $Max_i = (x_{maxi}, y_{maxi})$ e que o retângulo envolvente da janela de visão em $t_f = N_q - 1$ tenha coordenadas $Min_f = (x_{minf}, y_{minf})$ e $Max_f = (x_{maxf}, y_{maxf})$. A Figura 7.6(a) ilustra os dois retângulos envolventes. Daí tem-se que a área de carregamento é definida por seis vértices $Area_c = (v_0, v_1, v_2, v_3, v_4, v_5)$, onde $v_0 = (x_{mini}, y_{maxi})$, $v_1 = (x_{mini}, y_{mini})$, $v_2 = (x_{maxi}, y_{mini})$, $v_3 = (x_{maxf}, y_{minf})$, $v_4 = (x_{maxf}, y_{maxf})$ e $v_5 = (x_{minf}, y_{maxf})$ (Figura 7.6(b)).

No último caso, onde a câmera está em movimento desacelerado, é necessário encontrar o parâmetro t_z onde a câmera para de se locomover. Esse parâmetro é encontrado utilizando a equação da velocidade de movimento retilíneo uniforme, isto é, $v_f = v_i + at$. Fazendo $v_f = 0$, tem-se que $t = \frac{-v_i}{a}$. Aplicando essa equação em cada grau de liberdade da câmera, tem-se um conjunto $\mathcal{T} = (t_1, t_2, \dots, t_n)$, onde n é o número de graus de liberdade. O parâmetro t_z é definido como o menor elemento positivo do conjunto \mathcal{T} . Daí podem surgir duas situações. Na primeira, onde $t_z \geq N_q$, o problema fica reduzido à situação onde a câmera está em movimento acelerado (Figura 7.7(a)). Isto ocorre porque apesar da câmera

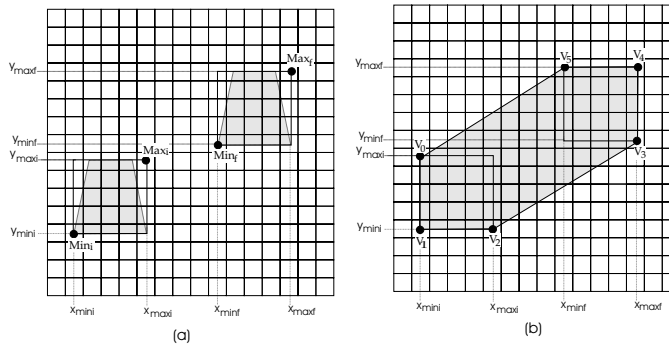


Figura 7.6: O cálculo da área de carregamento para o caso onde a câmera está em movimento acelerado;(a) Os retângulos envolventes das janelas de visão utilizados para determinar a área de carregamento; (b) As coordenadas dos seis vértices que definem a área de carregamento.

está desacelerando, a mudança de direção ocorrerá num quadro futuro que está além dos limites especificados para a predição. No segundo caso, onde $t_z < N_q$, é necessário verificar se o centróide da janela de visão obtida em $t = N_q - 1$ está antes do centróide da janela de visão obtida em $t = 0$. O centróide de uma janela de visão obtido em um quadro k é definido por $C_k = \frac{Min_k + Max_k}{2}$. A verificação é feita calculando o produto escalar entre o vetor $V_{N_q} = C_{t_{N_q-1}} - C_0$ e o vetor $V_c = C_{t_z} - C_0$. Se o produto escalar for maior igual a zero, então $t_i = 0$ e $t_f = t_z$ senão $t_i = N_q - 1$ e $t_f = t_z$. As Figuras 7.7(b) e (c) ilustram essas duas situações que podem ocorrer no caso onde $t_z < N_q$. Com os vetores de t_i e t_f basta utilizar o mesmo processo mostrado para o caso de movimento acelerado para obter a área de carregamento.

Escolha do Nível de Resolução

Após o cálculo da área de carregamento, é necessário determina o nível de resolução que deve ser carregado. Essa escolha é feita baseada em dois critérios: taxa de transmissão de dados e distância.

Inicialmente, é necessário determinar qual a resolução ideal que deve ser carregada dentro da área de carregamento. A área de carregamento é estimada a partir da janela inicial calculada em t_i e da janela final calculada em t_f . A resolução ideal para as duas janelas é obtida a partir da operação de busca `BestLevel`. Considere que o nível de resolução ideal para a janela inicial seja um nível N_i e para a janela final seja um nível N_f .

Suponha que a taxa de transmissão seja tx ladrilhos por quadro e que dentro da área de carregamento existem n_k ladrilhos no nível k . O

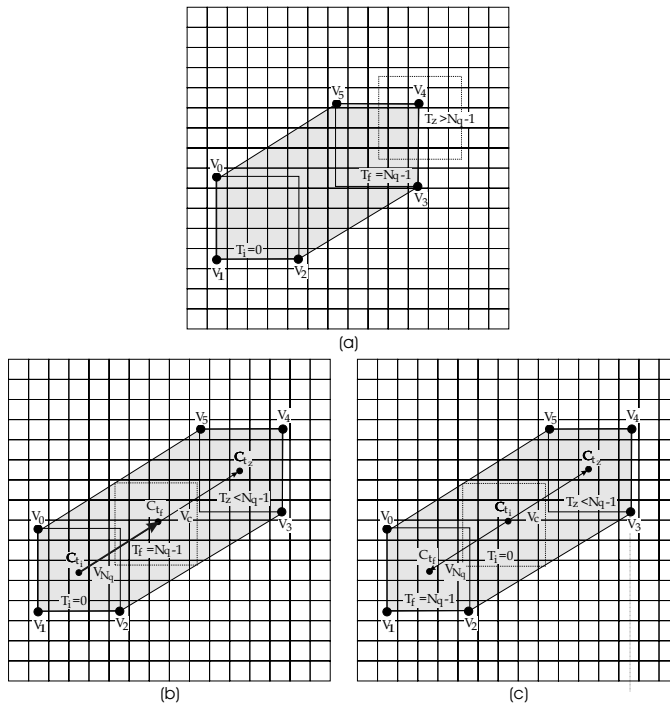


Figura 7.7: O cálculo da área de carregamento para o caso onde a câmera está em movimento desacelerado; (a) Ilustra o caso onde $T_z \geq N_q$; (b) Ilustra o caso onde a área é definida a partir dos quadros $t_i = 0$ e $t_f = t_z$; (c) Ilustra a situação onde a área é definida pelos quadros $t_i = N_q - 1$ e $t_f = t_z$.

nível de resolução escolhido é aquele que possuir a razão $R_k = \frac{n_k}{tx}$ mais próxima de 1. A Figura 7.8 mostra a quantidade de ladrilhos dentro da área de carregamento para cada nível de resolução de um objeto gráfico. Supondo que a taxa de transmissão seja de 25 *ladrilhos/quadro*, então $R_0 = \frac{4}{25} = 0.16$, $R_1 = \frac{11}{25} = 0.44$, $R_2 = \frac{27}{25} = 1.08$ e $R_3 = \frac{83}{25} = 3.32$. Assim, o nível escolhido por esse critério é o nível 2.

No critério de distância a área de carregamento é dividida em regiões. Considere que a área de carregamento seja definida pelos vértices $A = (v_0, v_1, v_2, v_3, v_4, v_5)$ e que os centróides das janelas inicial e final sejam C_i e C_f , respectivamente. A partir desses parâmetros são definidos três vetores unitários: $V_c = \frac{C_f - C_i}{\|C_f - C_i\|}$, $V_l = \frac{v_5 - v_0}{\|v_5 - v_0\|}$ e $V_r = \frac{v_3 - v_2}{\|v_3 - v_2\|}$.

A divisão é feita ao longo do eixo definido pelo vetor V_c . Suponha que cada região tem comprimento fixo de D_v na direção de V_c . As sub-regiões de carregamento são definidas por seis vértices, ou seja, $A_k = (v_0^k, v_1^k, v_2^k, v_3^k, v_4^k, v_5^k)$, onde $k < R$. O símbolo R representa o número de divisões. Os vértices da primeira e da última região são calculados de forma especial. Os vértices da primeira região A_0 são definidos por $v_0^0 = v_0$, $v_1^0 = v_1$, $v_2^0 = v_2$, $v_3^0 = v_2 + D_v \cdot V_r$, $v_4^0 = C_i + D_v \cdot V_c$, $v_5^0 = v_0 + D_v \cdot V_l$. A última região

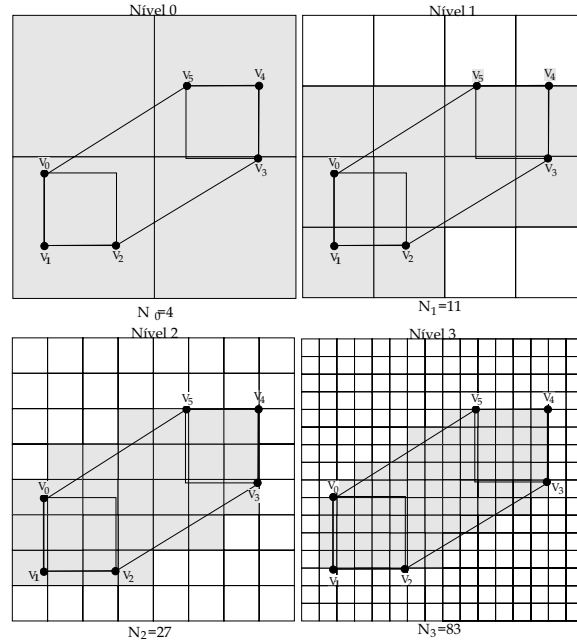


Figura 7.8: A figura mostra um objeto gráfico com quatro níveis de resolução. Os ladrilhos que estão dentro ou que interceptam a área de carregamento são marcados como candidatos para o carregamento. Isto é feito para cada nível de resolução.

A_{R-1} é definida pelos vértices $v_0^{R-1} = v_2 + R \cdot D_v \cdot V_r$, $v_1^{R-1} = C_i + R \cdot D_v \cdot V_c$, $v_2^{R-1} = v_0 + R \cdot D_v \cdot V_l$, $v_3^{R-1} = v_5$, $v_4^{R-1} = v_4$, $v_5^{R-1} = v_3$. As regiões intermediárias são calculadas de maneira única. Desta forma, uma região A_k , onde $0 < k < R - 1$, é definida pelos vértices $v_0^k = v_2 + k \cdot D_v \cdot V_r$, $v_1^k = C_i + k \cdot D_v \cdot V_c$, $v_2^k = v_0 + k \cdot D_v \cdot V_l$, $v_3^k = v_0 + (k + 1) \cdot D_v \cdot V_l$, $v_4^k = C_i + (k + 1) \cdot D_v \cdot V_c$, $v_5^k = v_2 + (k + 1) \cdot D_v \cdot V_r$. A Figura 7.9 mostra as sub-regiões de carregamento. Observe que a primeira e a última região são maiores que as regiões intermediárias. Isto ocorre porque a divisão é feita na região que está entre os centróides.

Suponha que o nível de resolução escolhido no critério de taxa de transmissão seja N_k . O comprimento total da área de carregamento ao longo do vetor V_c é $D_c = \|C_f - C_i\|$, assim o número de regiões R é dado por

$$R = \min \left(N, \left\lfloor \frac{D_c}{D_v} \right\rfloor \right).$$

O nível de resolução N é definido por $N = \min(N_k, N_i)$. As regiões são ordenadas de acordo com a sua distância em relação a posição atual da câmera. O nível de resolução escolhido para cada região é dado por $Rn_i = \min(N - i, N_f)$, onde $0 \leq i < R$.

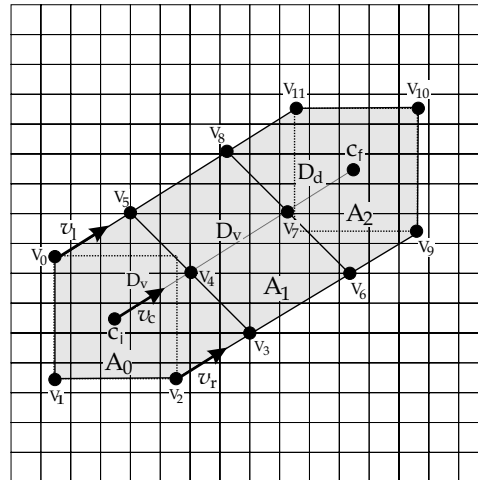


Figura 7.9: A área de carregamento é dividida em sub-regiões. Nesse exemplo a área de carregamento é dividida em três regiões: $A_0 = (v_0, v_1, v_2, v_3, v_4, v_5)$, $A_1 = (v_3, v_4, v_5, v_6, v_7, v_8)$ e $A_2 = (v_6, v_7, v_8, v_9, v_{10}, v_{11})$

Intuitivamente, o que se deseja é carregar apenas os ladrilhos que serão utilizados nos quadros futuros e não sobrecarregar o sistema com um número muito grande de pedidos. O critério de taxa de transmissão tem a função de evitar a sobrecarga do sistema. Quanto maior a velocidade da câmera maior será a área de carregamento e maior o número de ladrilhos que estão dentro dessa área. Assim, segundo esse critério um nível de menor resolução será escolhido. Esse efeito é desejado porque quanto maior a velocidade da câmera menor é a capacidade do usuário perceber detalhes da cena. Desta forma, não é necessário perder tempo carregando ladrilhos de alta resolução. Além disso, se o nível de resolução resultante do critério de taxa de transmissão for maior que o nível requerido pela aplicação, então o nível solicitado pela aplicação é o escolhido.

A probabilidade da câmera passar por determinada região que está dentro da área de carregamento diminui com a distância entre essa região e a posição atual da câmera. Assim, não é necessário perder tempo carregando ladrilhos de alta resolução em regiões que estão distantes da posição atual da câmera, pois a probabilidade deles serem efetivamente utilizados pela aplicação é muito pequena. O critério de distância tem a finalidade de simular esse fenômeno. Segundo esse critério, a área de carregamento é dividida em regiões que podem ser ordenadas segundo o fator de tempo. Assim, quanto maior o tempo para atingir uma determinada região menor é o nível de resolução que deve ser carregado. Da mesma forma que no caso

anterior, o nível de resolução que será carregado é o menor entre o nível escolhido pelo critério de distância e o requerido pela aplicação.

O sistema de predição utiliza a operação de busca `TileAddress` para obter os endereços dos ladrilhos que estão dentro das sub-regiões de carregamento. Assim, o sistema passa os vértices da área de carregamento, o nível de resolução e um vetor de endereços virtuais. Em resposta, o objeto gráfico inicia os elementos do vetor de carregamento com os endereços virtuais dos ladrilhos que estão dentro da área e no nível de resolução desejado. Em seguida, o vetor de endereços será passado para sistema de carregamento de página que providenciará que os endereços sejam carregados para o estágio mais alto de armazenamento.

Área de Descarregamento

A área de descarregamento tem o objetivo de corrigir erros cometidos no processo de predição. Esses erros estão relacionados com as mudanças inesperadas no padrão de variação dos parâmetros da câmera. Isto faz com que alguns de pedidos carregamento não sejam mais necessários. Desta forma, é necessário cancelar os pedidos de carregamento dos ladrilhos que ainda não foram carregados e que não fazem mais parte do conjunto de ladrilhos que serão utilizados num futuro próximo.

Matematicamente, a área de descarga é definida como área resultante da diferença entre a área de carregamento calculada no quadro anterior e a área de carregamento calculada no quadro atual, como mostra a Figura 7.10. Os pedidos de carregamento dos endereços dos ladrilhos que estão dentro dessa área são cancelados. A primeira solução para esse problema seria calcular a área de descarregamento e determinar os ladrilhos que estão dentro dela. No entanto, esse método é computacionalmente caro em relação ao tempo destinado para realizar essa tarefa.

A solução mais prática é identificar os ladrilhos que foram pedidos no quadro anterior e que não estão dentro da área de carregamento calculada no quadro atual. Observe que esse método dará o mesmo resultado que o método anterior, mas não foi necessário calcular explicitamente a área de descarregamento.

Os endereços virtuais dos ladrilhos pedidos no quadro anterior são obtidos a partir do vetor de carregamento. A operação cancelamento é realizada antes da operação de carregamento. Assim, os elementos do vetor de carregamento ainda armazenam os endereços virtuais dos ladrilhos pedidos no quadro anterior. A classificação dos ladrilhos é feita a partir

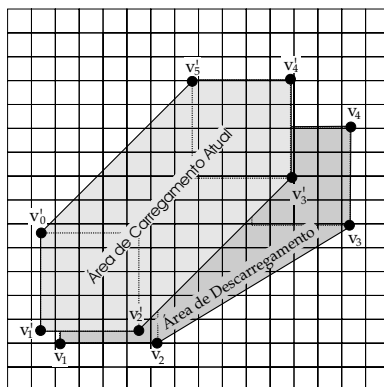


Figura 7.10: Os pedidos de carregamento dos ladrilhos que estão dentro a área de descarregamento são cancelados.

da operação de busca `TileClassify`. Na operação de busca são passados como parâmetros um ladrilho e a área de carregamento do quadro atual. O ladrilho é obtido a partir da conversão do endereço virtual em um nó da quad-tree. A conversão é feita utilizando a tabela de mapeamento existente dentro da estrutura `ObjectMemory`. Em resposta, o objeto gráfico informa se o ladrilho está dentro ou fora da área de carregamento. Se o ladrilho estiver fora da área, então o seu endereço virtual é inserido no vetor de cancelamento.

O vetor de cancelamento é passado para o sistema de carregamento. Os pedidos de cancelamento são replicados até o estágio de armazenamento onde os dados associados ao endereço virtual ainda não foram carregados. A operação de cancelamento é realizada segundo as regras de liberação de páginas. Essas regras são apresentadas na Seção 7.2.1.

7.1.2 Regras de Carregamento de Páginas

O sistema de regras de carregamento é responsável por determinar os requisitos necessários para que uma página de dados seja carregada. O sistema de regras está ligado à unidade de controle. Desta forma, os pedidos de carregamento de páginas são realizados diretamente pelo sistema de regras.

A interface projetada para esse sistema é composta por um operador para carregar as páginas de dados, um operador para especificar uma unidade de controle e um operador de inicialização. O operador de carregamento implementa as regras de carregamento de páginas. O operador de inicialização é chamado pelo sistema de gerenciamento de memória após

a operação de carregamento do objeto gráfico. Esse operador implementa as ações iniciais que devem ser tomadas pelo sistema de regras.

As regras foram elaboradas para objetos gráficos representados em multi-resolução. Essa representação é codificada a partir de uma floresta de “quad-trees”. As regras para carregamento são as seguintes:

Regra C1. Para um objeto gráfico ser visualizado é necessário que as páginas das texturas do nível de mais baixa resolução sejam carregadas. Assim, é assumido que o estágio de armazenamento deve ser capaz de armazenar pelo menos o nível mais baixo de resolução da textura. A Figura 7.11 ilustra essa regra.

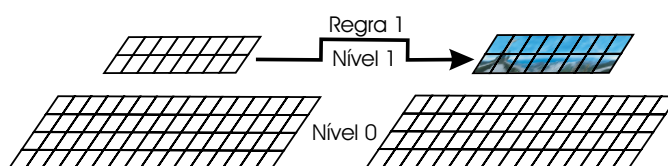


Figura 7.11: Regra C1: O nível mais baixo de resolução deve ser carregado na memória.

Regra C2. Se a textura de um ladrilho é carregada, as texturas dos seus irmãos também são. Essa regra é naturalmente atingida visto que uma página armazena as texturas dos ladrilhos que são filhos do mesmo ladrilho pai. Essa regra é mostrada na Figura 7.12.

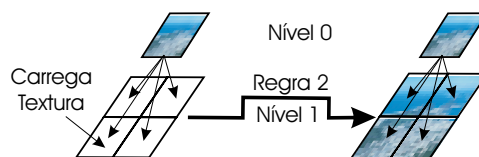


Figura 7.12: Regra C2: Se a textura de um ladrilho for carregada, então as texturas dos seus irmãos também devem ser carregadas.

Regra C3. Se a textura de um ladrilho que está no nível k for carregada, então as texturas dos ancestrais desse ladrilho também deverão ser carregadas. A Figura 7.13 ilustra o resultado dessa regra.

A primeira regra é implementada a partir do operador de inicialização. A segunda regra é atingida naturalmente porque as texturas de ladrilhos irmãos são armazenadas na mesma página de dados. A terceira regra é implementada no operador de carregamento. A implementação dessa última

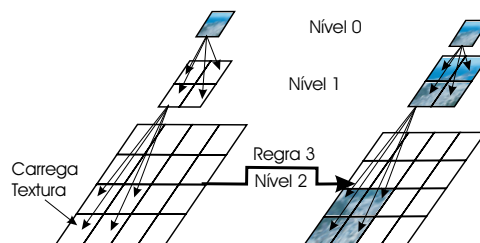


Figura 7.13: Regra *C3*: Se a textura de um ladrilho que está em um determinado nível for carregada, então as texturas dos seus ancestrais também devem ser carregadas.

regra é realizada a partir da tabela de mapeamento que existe dentro da estrutura `ObjectMemory`, veja o capítulo anterior. Essa tabela é utilizada para converter o endereço virtual em um nó da árvore. A conversão permite caminhar na quad-tree. Assim, é possível recuperar os endereços virtuais dos ladrilhos ancestrais de um determinado ladrilho.

O sistema de visualização envia os parâmetros da câmera utilizados para sintetizar as cenas. Utilizando esses parâmetros, o sistema de predição calcula os valores dos parâmetros da câmera para as próximas k cenas. Para cada parâmetro de câmera estimado, o sistema de carregamento de páginas realiza duas operações:

1. Pede para o objeto gráfico retornar os endereços virtuais dos ladrilhos que estão dentro do frustum de visão.
2. Pede para carregar as páginas que contém os dados dos ladrilhos. Esses pedidos são realizados obedecendo as regras *C2* e *C3*.

7.2

Sistema de Liberação de Páginas

A remoção de páginas só é necessária quando o estágio de armazenamento está cheio. Nesse caso, ativa-se o mecanismo de substituição de páginas. Esse mecanismo chama o sistema de liberação para decidir qual página deve ser retirada da memória. O espaço liberado é utilizado para armazenar uma nova página. A figura 7.14 mostra a arquitetura do sistema de liberação.

O sistema é formado por dois componentes: a estrutura `ObjectMemory` e o subsistema de regras. O objeto gráfico tem a função de calcular a prioridade de uma página de textura que está armazenada no estágio de armazenamento. A comunicação com o objeto gráfico é feita a partir

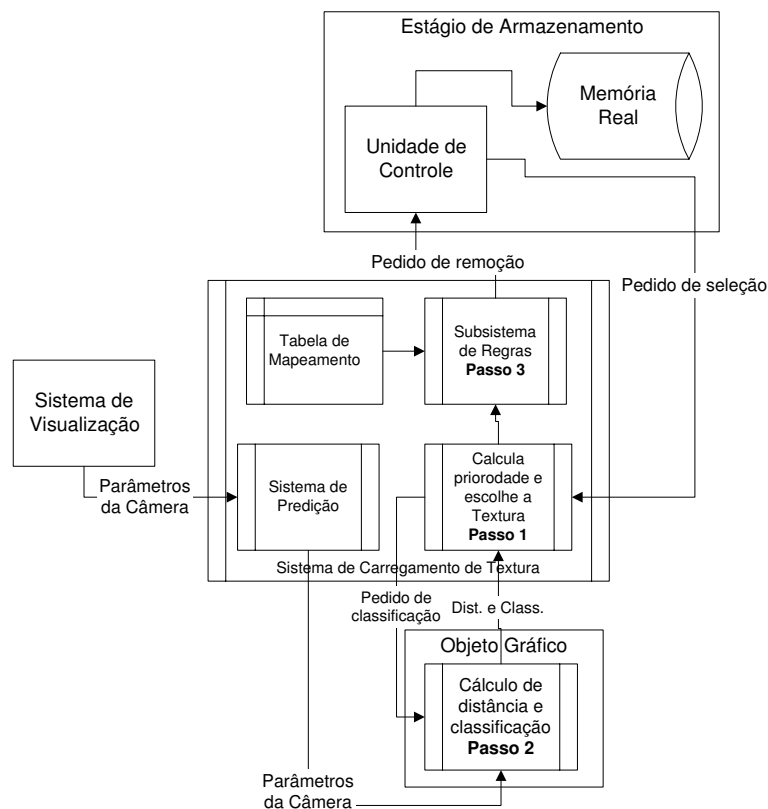


Figura 7.14: Arquitetura e funcionamento do sistema de liberação de texturas.

do operador de busca *TileClassify*. Ao chamar esse operador, o sistema passa para o objeto gráfico um ladrilho e a área de carregamento. Em resposta, o objeto gráfico informa o estado do ladrilho em relação à área de carregamento (dentro, fora ou intercepta) e a distância entre o centróide do ladrilho e o centroide da área de carregamento. A partir dessas informações o sistema calcula uma prioridade para a ladrilho dentro intervalo $[0, 1]$. O ladrilho que receber a menor prioridade é liberado do estágio de armazenamento.

Apenas os ladrilhos considerados folhas são avaliados pelo sistema. Um ladrilho é considerado folha se a textura de todos os ladrilhos descendentes não estiverem carregadas no estágio de armazenamento.

No critério de distância quanto mais longe o ladrilho estiver da câmera menor é a sua prioridade (Figura 7.15(a)). No critério de profundidade quando maior o nível de resolução menor a sua prioridade (Figura 7.15(b)). O valor da prioridade é definido a partir da média ponderada desses dois critérios.

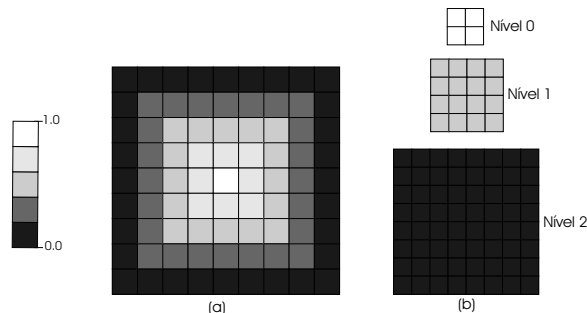


Figura 7.15: Os critérios utilizados para determinar a prioridade de um ladrilho. (a) Classificação dos ladrilhos baseado na critério de distância; (b) Classificação dos ladrilhos baseado no critério de profundidade; A paleta de cor mostra o significado das cores.

Suponha que o suporte paramétrico seja definido por $[a, b] \times [c, d]$, que a área de carregamento seja definida pelos vértices $A = (v_0, v_1, v_2, v_3, v_4, v_5)$ e que um ladrilho seja definido por $[la, lb] \times [lc, ld]$, como mostra a Figura 7.16. O centro do ladrilho é dado por $Centro_l = (cx_l, cy_l) = (\frac{la+lb}{2}, \frac{lc+ld}{2})$ e o centroide da área é dado por $Centro_a = (cx_a, cy_a) = \frac{v_0+v_1+v_2+v_3+v_4+v_5}{6}$. A distância máxima entre um ladrilho e uma janela que está dentro do suporte paramétrico é dada por $MaxDist = \sqrt{(a-c)^2 + (b-d)^2}$. Assim, o valor da prioridade de um ladrilho é definido por $Prio_d = 1 - \frac{\sqrt{(cx_l-cx_a)^2 + (cy_l-cy_a)^2}}{MaxDist}$. A informação de classificação de ladrilho é utilizada para atenuar ou aumentar a prioridade $Prio_d$. Se o ladrilho estiver dentro da área de carregamento então a sua prioridade é $Prio_c = (1 - \beta)Prio_d$, caso contrário a sua prioridade é $Prio_c = \beta Prio_d$. O valor da constante β deve estar no intervalo $[0, 0.5]$. Observe que se $\beta = 0.5$ a operação de classificação não é levada em consideração.

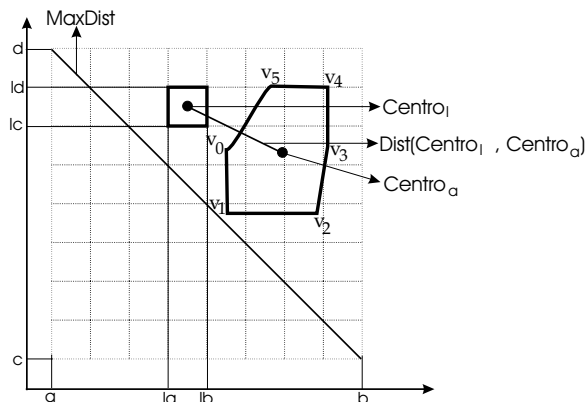


Figura 7.16: Visualização das variáveis utilizadas no cálculo da prioridade baseada na distância.

Supondo que a representação em multi-resolução tenha no máximo K níveis e que um ladrilho esteja no nível n , tem-se que o valor da sua prioridade em relação ao critério de profundidade é dado por $Prio_p = 1 - \frac{n}{K}$. Desta forma, a prioridade final associada a um ladrilho é dada por $Prioridade = (1 - \alpha)Prio_p + \alpha Prio_c$, onde $0 \leq \alpha \leq 1$. Intuitivamente, a constante α determina qual dos dois critérios terá maior influência no cálculo da prioridade de um ladrilho.

No sistema foram implementados dois algoritmos para a escolha da melhor página a ser retirada. O primeiro algoritmo escolhe a página de textura que tiver menor prioridade. O segundo escolhe a primeira página de textura cuja prioridade seja menor que um determinado valor. A vantagem do primeiro algoritmo é que é mais preciso que o segundo. A desvantagem é que é mais demorado, pois é necessário varrer toda a tabela de páginas do estágio de armazenamento. Os testes mostraram que é melhor utilizar o primeiro algoritmo quando o estágio tem pouca capacidade de armazenamento (menos páginas de dados) e para o caso contrário é melhor utilizar o segundo. O tipo do algoritmo é especificado durante a criação do sistema.

7.2.1

Regras de Liberação de Páginas

O subsistema de regras de liberação implementa as regras definidas para gerenciar as operações de remoção de páginas de texturas. Essas regras foram criadas levando em consideração que o objeto gráfico representa a sua textura em multi-resolução a partir de uma floresta de quad-trees e que a aplicação realiza operações de visualização sobre esses objetos.

As regras de liberação foram elaboradas para que as operações de remoção e carregamento de texturas sejam executadas de forma consistente. Assim, a principal finalidade das regras tanto de remoção quanto das regras de carregamento é manter a estrutura em multi-resolução sem buracos, isto é, se um ladrilho de um determinado nível de resolução está com os seus dados carregados, então todos os seus ancestrais também estão. Essa propriedade é verdadeira em todos os estágios de armazenamento. As regras de liberação de páginas são as seguintes:

Regra L1. As páginas que contêm as texturas do nível mais baixo de resolução não podem ser liberadas. Essa regra evita inconsistência durante a execução da primeira tarefa realizada quando ocorre uma ausência de páginas.

Regra L2. A textura de um ladrilho pai só poder ser liberada depois das texturas dos seus quatro filhos, veja a Figura 7.17. Essa regra também evita inconsistência durante a execução da primeira tarefa realizada quando ocorre uma ausência de páginas, veja próxima seção.

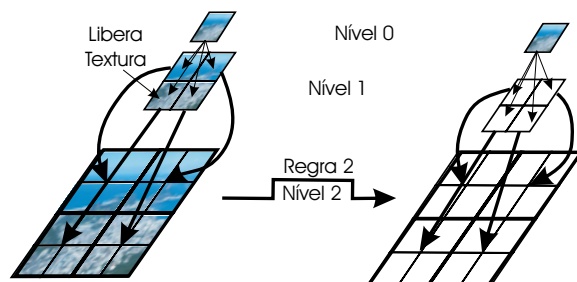


Figura 7.17: Procedimento realizado pelo sistema de liberação de páginas quando a textura de um ladrilho deve ser liberada da memória.

Regra L3. As páginas que contém as texturas que fazem parte do quadro corrente não podem ser liberadas.

7.3 Tratamento de Ausência de Páginas

O sistema de tratamento de ausência de páginas é responsável por lidar com os acessos aos dados que não foram carregados no estágio. Os casos de ausência de páginas podem ocorrer devido a três motivos:

1. O tempo destinado ao processamento do sistema pode não ser suficiente para carregar todas as páginas necessárias.
2. O erro cometido no processo de predição. Esse erro pode ocorrer quando os parâmetros da câmera variam bastante no decorrer do tempo.
3. Não existe espaço suficiente no estágio mais alto de armazenamento para armazenar as páginas necessárias para sintetizar os quadros futuros.

Como já foi discutido, o sistema de carregamento de páginas não pode interromper o processo de visualização para carregar uma página ausente. Assim, esse problema é resolvido explorando a estrutura de multi-resolução do objeto gráfico. Após receber o sinal da unidade de controle, o sistema de ausência de páginas realiza as seguintes tarefas:

1. Decodifica o identificador da página e do objeto gráfico a partir do endereço virtual.
2. Acessa a tabela de mapeamento e converte o endereço virtual em um ladrilho da árvore.
3. Varre a árvore de multi-resolução a procura de um ancestral desse ladrilho que está com os seus dados carregados. Esse ladrilho sempre será encontrado devido a Regra *C1*.
4. Retorna o endereço real dos dados do ladrilho encontrado e o nível de resolução a que o dado pertence.

A Figura 7.18 ilustra todos os passos realizados pelo sistema.

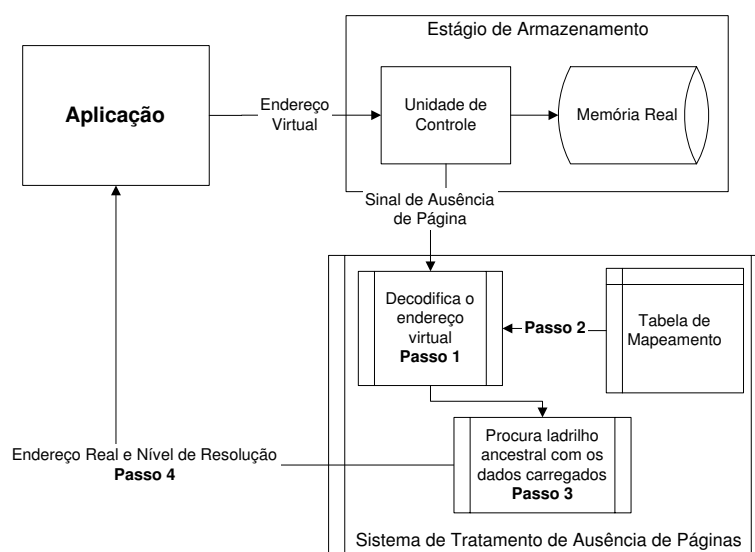


Figura 7.18: Os passos realizados pelo sistema de tratamento de ausência de páginas.

A informação do nível de resolução é utilizada para o objeto gráfico calcular as coordenadas de textura que mapeiam corretamente a geometria do ladrilho na textura retornada pelo sistema. A região mapeada representa a textura em baixa resolução do ladrilho. A Figura 7.19 mostra como o resultado desse mapeamento. O cálculo da função de mapeamento de textura de um ladrilho em relação as texturas de todos os seus ancestrais pode ser realizado na fase de pré-processamento. Esse cálculo é realizado a partir da modificação do Algoritmo 4 apresentado no Capítulo 4. Esse artifício é utilizado para tornar mais rápido o processo de visualização.

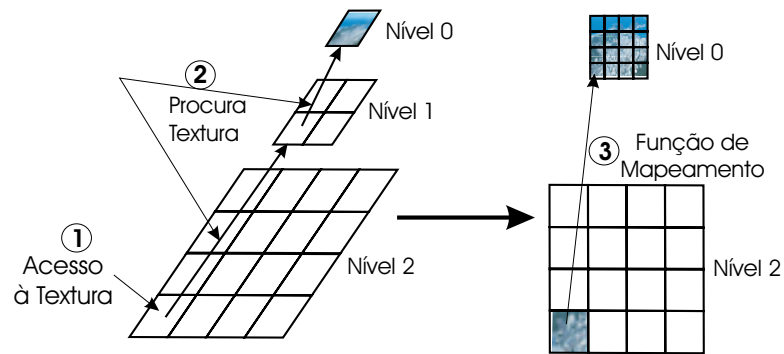


Figura 7.19: Procedimento realizado pelo sistema de carregamento de páginas quando ocorre um acesso a uma página ausente.