Laboratório VISGRAF
Instituto de Matemática Pura e Aplicada

Video de Quarta Geracao - Memoria 2008

Luiz Fernando Cordeiro
Luiz Velho (orientador)

Technical Report TR-2009-02 Relatório Técnico

January - 2009 - Janeiro

IMPA – Associação Instituto Nacional de Matemática Pura e Aplicada

Memória de Projeto – Vídeo de Quarta Geração

Bolsista: Luiz Fernando Magalhães Cordeiro
Período: Setembro de 2007 a Agosto de 2008

O projeto de Vídeo de Quarta Geração visa obter vídeos com informações sobre a profundidade dos objetos filmados. Para isso é necessário que a captura seja feita de forma a coletar dados que nos permitam estimar, com certa precisão, informações sobre a posição e o formato 3D dos objetos em cena.

Hardware Utilizado

- Dispositivo projetor de imagens (projetor DLP);
- Interface de saída de vídeo sincronizável (placa gráfica com entrada de sincronismo vertical);
- Computador para a geração das imagens (faixas BCSL) a projetar;
- Dispositivo de captura de imagens (câmera, comum, ou com saída DV Firewire);
- Computador com placa capturadora de vídeo ou com entrada DV Firewire.

Observações:

O primeiro computador, responsável pela geração das faixas a projetar, pode ser substituído por um hardware específico, implementado usando-se microcontrolador e elementos de reconhecimento e geração de sinais SVGA.

O outro computador pode ser de uso geral, uma vez que a maior parte do trabalho será feito pela placa de captura ou pela interface DV. É interessante, porém, que ele tenha boa capacidade de processamento, em geral, para que possa fazer, opcionalmente, a compressão em tempo real do vídeo a gravar; e para que possa ter um bom tempo de resposta na etapa de reconstrução do vídeo capturado.

Software Necessário

Todo o sistema foi implementado em C++, na plataforma Windows (Windows XP). Como ambiente de programação e depuração, utilizou-se: Visual Studio .NET, Dev C++ e MinGW. Como interface, usou-se: a própria API do Windows, o DirectShow, o OpenGL, o FLTK e o IUP (não no mesmo módulo, é claro).

Ambientação

Conforme pode-se imaginar, devido às diversas tecnologias envolvidas (tanto de hardware como de software, bem como tecnologias específicas relacionadas à captura e conversão de vídeo), uma etapa de ambientação e aprendizagem é fundamental para se situar no projeto. Antes mesmo de se familiarizar com os códigos dos módulos já desenvolvidos, torna-se necessário um domínio mínimo das tecnologias utilizadas. Após um período de aprendizado inicial, deve-se passar ao estudo dos sistemas criados, retornando frequentemente ao estudo das tecnologias, para o aprofundamento necessário.

Durante esse período de reconhecimento e estudo dos códigos, pode-se ter, eventualmente, uma visão crítica dos métodos utilizados e abordagens assumidas. Nesta etapa, um questionamento sobre se tal ou qual tecnologia é mais adequada torna-se parte da rotina de estudo. As soluções encontradas são muitas vezes, pode-se dizer, brilhantes. Outras vezes, porém, refletem apenas um desconhecimento ou inexperiência em determinada área da computação.

Deve-se dizer, contudo, que existe o outro lado da questão, qual seja: a própria ignorância daquele que toma pé do projeto pode fazer com que sua visão crítica seja inadequada, e suas conclusões sejam precipitadas e totalmente (ou parcialmente) erradas. O estudo mais aprofundado das implementações, e das potencialidades e limitações dos sistemas utilizados, leva à correção dessa visão, inicialmente imatura.

Descrição do Sistema

A utilização do sistema pode ser decomposta nas etapas seqüenciais a seguir:
1 - Ajustes de câmera e projetor, com calibrações cromáticas e geométricas;
2 - Captura de vídeo, com projeção das faixas BCSL, gerando arquivo AVI.
3 - Correções cromáticas do vídeo capturado;
4 - Pré-processamento do arquivo capturado, com correção cromática e geração de quadros adicionais de textura, a partir dos quadros com faixas, gerando novo arquivo AVI;
5 - Processamento do arquivo corrigido, detectando-se as fronteiras entre as faixas, gerando arquivo específico com informações de profundidade do vídeo.

Etapa Anterior

Nas etapas anteriores do projeto (antes de 2007), as etapas acima estavam agrupadas em apenas dois módulos: um responsável pela calibração cromática, captura, e processamento (tudo em tempo real); e outro responsável apenas pela calibração geométrica.

Obteve-se uma baixa taxa de amostragem: apenas 5 quadros por segundo. A resolução geométrica (das faixas) chegou a 90 faixas, porém com uma quantidade de erros geométricos muito alta.

Etapa Atual

Nesta nova etapa de trabalhos, a partir de meados de 2007, partiu-se para uma avaliação das possíveis causas dos problemas encontrados, bem como o estabelecimento de estratégias de correção dos mesmos e até de mudanças de abordagens de trabalho.

Medições de tempo mostraram que o hardware utilizado não era capaz de executar o processamento da captura em tempo real. Embora a máquina utilizada dispusesse de mais de um processador, estes não podiam trabalhar em conjunto. O tamanho de cada frame (cerca de 1 Megabytes), 30 vezes por segundo, e com várias cópias, fazia com que a memória cache fosse continuamente atualizada, levando o processador a trabalhar muito abaixo de sua capacidade máxima.

Embora a capacidade do hardware não suportasse captura e processamento simultâneos, ambos os processos poderiam ser executados separadamente. Somente a captura tinha restrições em relação à sincronia com câmera e projetor. Os processamentos de correção cromática e de detecção geométrica poderiam ser executados posteriormente, usando como entrada um arquivo gravado a partir da captura. A gravação do arquivo levava, porém, à outra restrição do hardware: a taxa de gravação em disco. Esta ficava em torno de 30 Megabytes por segundo (720h x 480v x 3c x 30f).

Embora a máquina utilizada fosse capaz desta taxa de gravação, observou-se que, eventualmente, quadros eram replicados pelo DirectShow. Isso era feito, aparentemente, como estratégia de manutenção da taxa fixa de amostragem e gravação. Supõe-se que, devido à nossa incapacidade visual de distinguir diferenças pequenas de um quadro para outro, os projetistas do DirectShow tenham optado por gravar dois quadros iguais caso a fila de saída (gravação) do sistema estivesse cheia. Uma vez que a gravação em disco praticamente não gasta tempo de processador, por ser feita pelo D.M.A., a replicação de um quadro seria uma solução viável para o problema de congestionamento da fila, não levando a efeitos visuais relevantes.

Isto é ótimo para vídeos típicos, mas péssimo para nossa aplicação. Nela, cada campo de cada quadro carrega informações que são, em princípio, únicas. A replicação de um quadro faz com que se perca dados de dois quadros.

Observou-se que, mesmo retirando-se o processamento de detecção de geometria, o processador ainda ficava sobrecarregado, levando ao problema acima. Verificou-se que parte desse processamento excessivo era causado pela própria interface com o usuário. A visualização dos quadros capturados, em tamanho real, gerava, no Sistema Operacional, uma sobrecarga de trabalho na atualização da janela. A solução adotada foi reduzir-se a 1/16 (1/4 x 1/4) o tamanho da amostra a visualizar. Com a sobra de processamento alcançada, passou-se a mostrar quatro quadros (8 campos) de cada vez, na tela, na taxa de 7,5 amostras por segundo.

Essa última modificação teve desdobramentos inesperados!

A iluminação da cena capturada é feita usando-se dois conjuntos duplos de faixas: dois slides. Cada slide é mostrado no tempo de um único campo (1/60 s). Seu complementar cromático é mostrado em seguida. Isso leva a se mostrar quatro diferentes conjuntos de faixas em cada grupo. Com a visualização de quatro quadros simultâneos, consegue-se ter uma visão "parada" das faixas projetadas.

Superado o problema de perda de quadros na gravação, atacou-se o problema do aumento da quantidade de grupos de slides por segundo. Embora seja possível, em princípio, restaurar a informação de textura da cena a partir de processamentos de grupos de faixas complementares, verificou-se que, na prática isso gerava imagens de muito baixa qualidade. Na época das primeiras implementações não se chegou a determinar a origem de tal problema. Mesmo corrigindo-se as distorções cromáticas (possível causa do problema) não se conseguiu bons resultados visuais. Optou-se, na época, por reduzir-se a taxa de grupos de slides, acrescentando-se um quadro com iluminação branca (sem faixas). Isso reduziu a taxa de grupos, de 15 para 10 por segundo.

No novo sistema, tentou-se novas formas de geração dos grupos. Uma das possíveis melhorias seria projetar um slide de faixas, seguido de um slide com iluminação branca. A textura seria obtida diretamente do slide "branco". As faixas complementares seriam obtidas da diferença entre pixels brancos e de cor (faixas). O intercalamento de um slide sem iluminação (preto) permitiria remover, pela diferença com o branco, a iluminação ambiente (iluminação não-projetada).

Em princípio, deveria funcionar. Mas não funcionou. Verificou-se que os canais de cor (vermelho, verde e azul) quase nunca eram puros. Por exemplo, uma iluminação somente pelo canal verde gerava luz vermelha. Um píxel iluminado com luz branca (R, G e B) nunca era o mesmo que um pixel com a soma de R, G e B de outros campos. A calibração cromática deveria corrigir este problema, mas não foi capaz. Aparentemente, a "mistura" de canais de cor não era linear; e os procedimentos de calibração cromática e de captura deveriam ser lineares. Esse tipo de efeito gerava slides complementares bastante distorcidos, com baixa qualidade para a detecção geométrica.

Verificou-se também, de forma inesperada, que o slide sem iluminação (preto) não ficava preto! Ele tinha faixas vermelhas! Havia "vazamento" de cor, de um campo (slide) para o seguinte. Isso foi detectado no slide preto pois era o que permitia maior visualização desse efeito. ´É claro que também deviam ocorrer vazamentos entre os outros campos. Isso foi confirmado, logo depois, em experimento próprio. Essa detecção só foi possível devido à visualização simultânea de vários campos.

O responsável por este efeito (defeito) inesperado era o projetor DLP. A projeção de campos, embora sincronizada com o sinal vertical, gerado pela câmera, não estava em fase. Havia, sempre, um tempo de cerca de 1 milissegundo entre o sinal vertical e o início da projeção dos campos. Para corrigir este efeito, tivemos que gerar um atraso de 16 milissegundos na placa de vídeo, compensando o atraso gerado pelo projetor. Mesmo assim, a projeção não ficou totalmente sincronizada, pois um projetor DLP usa uma rodinha de filtros

de cor para selecionar qual canal está sendo projetado. Como é uma coisa mecânica, gera uma pequena variação de fase, mesmo sendo sincronizada eletronicamente.

A solução para mais este problema foi mudar o atraso na placa de vídeo, para 4 milissegundos, e reduzir o tempo de exposição (shutter) da câmera, de 1/60 para 1/120. Isto fazia com que a transição entre slides, gerados pelo projetor, ficasse "longe" do intervalo de captura da câmera.

Testou-se um projetor DLP com 3 LEDs (ao invés de lâmpada e rodinha). A qualidade de cor foi sensivelmente superior, com pouquíssima mistura entre os canais. Infelizmente a baixa luminosidade dos LEDs limitava, de forma proibitiva, sua utilização para cenas normais, envolvendo pessoas. Somente a captura de imagens pequenas poderia ser feita com tal projetor.

Após todos os ajustes e calibrações descritos acima, passou-se a testar a capacidade de captura geométrica de faixas pelos módulos já criados. Para se ter um padrão de referência, criou-se um programa que gerava, sinteticamente, arquivos com imagens, de faixas de cores e geometria, perfeitas. O módulo de detecção de fronteiras de faixas reconheceu, perfeitamente, até uma grande quantidade de faixas (cerca de 200). O desempenho com projeção e captura por câmera ficou, contudo, muito aquém disso: cerca de 60 somente. Se aumentássemos o número de faixas a quantidade de erros de detecção crescia muito, degradando rapidamente a qualidade dos dados processados.

Uma das causas deste problema era que o sinal de vídeo, gerado pela câmera, era codificado em DV. A codificação DV usa uma melhoria da codificação JPG, que usa uma codificação DCT para blocos de 64 pixels de cada quadro. Esta forma de comprimir sinais de vídeo é aceitável visualmente, pois explora a natureza gradual de transições de luminância e crominância de uma imagem visualmente agradável. Verificou-se experimentalmente (há bastante tempo), que o olho humano é muito mais sensível às variações de luminância do que às de crominância. A compressão JPG, e a DV, usam esta característica para reduzir a taxa de dados gerado no processo de compressão. Faz-se uma "sub-amostragem" de crominância, ou seja, variações abruptas de cor são "suavizadas" no processo de codificação.

Isso não causa problemas num vídeo comum, pois nossos olhos não são capazes de perceber variações rápidas de cor. O mesmo não é verdade para o processo de detecção de geometria, baseado na projeção de faixas BCSL. Aqui, as transições abruptas de cor são a informação desejada!

Essa limitação, gerada pela codificação do sinal de vídeo, só poderá ser resolvida se usarmos câmeras com sinais de saída em três canais RGB separados, sem compressão. Ainda não fizemos testes com uma câmera desse tipo; espera-se, contudo, que o problema acima seja solucionado a contento, usando-se tal dispositivo.

Além da redução da "velocidade de transição" introduzida pela codificação DV, conseguimos detectar outro problema no sistema. Uma análise do perfil de sinais que chegavam ao programa detector mostrou que variações espúrias nos valores dos canais de cor "confundiam" o algoritmo de detecção. A aparência do sinal era de transições múltiplas nas fronteiras de faixas. Verificou-se que essas variações "falsas" não ocorriam em todas as fronteiras, mas somente naquelas em que mais de uma cor mudava, entre faixas, e somente quando uma das cores "subia" e outra "descia" em valor de luminosidade. O problema estava, aparentemente, na mistura de cores dos canais: quando um deles subia de valor, "carregava" outro que estava descendo. Este último, então, começava a descer, subia um pouco e descia novamente. Esta "indecisão" no canal confundia completamente o algoritmo de detecção.

Como o problema só acontecia em mudanças múltiplas de canal, gerou-se um novo código BCSL em que somente um canal pode mudar em cada transição de faixas. Este novo código resolveu completamente o problema. O único efeito negativo foi que a quantidade de transições de faixas possíveis, com esta restrição, foi reduzida, de 900 para 144. O sistema então ficou limitado a um máximo de 144 faixas. O novo código gerado, por questões de

compatibilidade com o programa de detecção, continua com 900 faixas possíveis, porém somente as primeiras 144 são com a garantia de transição num único canal.

Um aspecto ainda não mencionado foi a modularização dos sistemas. Na etapa anterior, os sistemas criados tinham um baixo índice de modularidade: foram criados e depurados de forma muito integrada. Uma das abordagens adotadas nesta nova etapa foi a de modularizar ao máximo o projeto, gerando uma melhor definição de interface entre os sub-sistemas, de forma a se ter um melhor controle sobre processos do sistema. Parte do trabalho desenvolvido foi o de desmembrar partes do código fonte, encapsulando-os em objetos bem definidos. Durante esses trabalhos, como é de se esperar, erros de implementação são detectados (e corrigidos).

Propostas e previsões

Espera-se, nesta nova etapa (2008-2009), obter melhorias no sistema na parte geométrica. Para tanto, torna-se fundamental expor-se o sistema a testes reais de utilização. Outros pesquisadores serão envolvidos no projeto, gerando demandas e detectando eventuais mal funcionamentos (erros de mais difícil detecção). Novos dispositivos de hardware deverão ser testados, tanto para corrigir alguns dos problemas relatados acima quanto para melhorar as definições geométrica e cromática. Outros algoritmos de detecção deverão ser testados. Novas interfaces com o usuário deverão ser implementadas e testadas. E uma conversão do sistema para a plataforma Linux deverá ser feita.

Glossário:

API	- Application Programming Interface
AVI	- Audio Video Interleaved
BCSL	- Binary Coded Structured Light
DCT	- Discret Cosine Transform
DLP	- Digital Light Processing
DV	- Digital Video
Fireware	- Interface serial rápida (padrão IEEE 1394)
FLTK	- Fast Light Tookit
IUP	- Portable User Interface
LED	- Light Emitting Diode
MinGW	- Minimalist GNU for Windows
MMR	- Micro Mirrors Device
OpenGL	- Open Graphics Library
PWM	- Pulse Width Modulation

Apêndices

1 - Modo de Trabalho de um Projetor DLP

Esta classe de projetores utiliza dispositivos com micro espelhos (MMR) para refletir (ou não) a luz de uma lâmpada de cor branca. O dispositivo é formado por uma matriz de espelhos que podem ser, cada um deles, rodados separadamente.
Uma única lâmpada emiti luz, continuamente, sobre o dispositivo, num determinado ângulo de incidência. Os espelhos tem duas posições possíveis somente: uma irá refletir a luz da lâmpada para uma lente de saída; a outra irá refletir a luz para um anteparo preto dentro do projetor. Como cada espelho pode ser acionado individualmente, uma "imagem reflexiva" é gerada, eletronicamente, na matriz, a cada quadro projetado.

Para se obter tons de cinza, faz-se uma modulação PWM de cada espelho. Todos os espelhos representando pixels não-pretos serão ativados simultaneamente. Mas cada um deles será desativado num tempo diferente, proporcional à luminosidade do pixel projetado.

Para se conseguir imagens coloridas, faz-se, ao invés de uma, três projeções de cada quadro. Para cada uma delas, um filtro cromático bloqueia dois canais de cor. A luz qua irá incidir sobre o dispositivo DLP será, então, em cada projeção: Vermelha, Verde e Azul. Os três filtros são colocados sobre uma rodinha, que fica rodando continuamente, sincronizada com a projeção dos quadros, de tal forma que um dos filtros estará na posição de bloqueio durante o tempo de reflexão de cada um dos três sub-quadros cromáticos.

Nos primeiros projetores, somente um conjunto (1x) de três sub-quadros era projetado. Isso gerava alguns efeitos visuais indesejáveis. Os projetores atuais são de, pelo menos, 4x, ou seja, para cada quadro, são feitos 4 conjuntos de 3 projeções.

Existem projetores DLPs que não utilizam a rodinha de filtros. Neles, ao invés de uma única lâmpada, são usados 3 LEDs: um vermelho, outro verde e outro azul. A qualidade cromática de projeção é superior ao do projetor com uma lâmpada, porém a luminosidade dos LEDs é muito inferior à da lâmpada. Somente podem ser usados para distâncias pequenas e com baixa iluminação ambiente.