Geometria e Programação Geométrica

Capítulo 5 (Foley & van Dam)

Carlos Hitoshi Morimoto (hitoshi@ime.usp.br)

Exemplos

Intersecções geométricas:
- Como sabemos quando uma bola bate na parede?

Orientação
- Como fazer com que um F1 aponte para os boxes?

Transformação
- como desenhar o carro em diferentes instantes?

Mudança de coordenadas
- como desenhar as rodas, uma de cada vez?

Reflexão e refração

Programação geométrica

Existem muitas áreas da Ciência da Computação que exigem o uso de entidades geométricas
- ✓ Computação gráfica
- ✓ CAD - computer aided design
- ✓ Robótica
- ✓ Visão computacional
- ✓ GIS - geografic information systems

Programação independente de coordenadas

- Tradicionalmente um livro de CG introduz um sistema de coords, vetores, matrizes, seguido por fórmulas que envolvem diversas matrizes 4x4.
- Embora conveniente, se torna difícil "ver" o que estamos programando.

Geometria e CG

CG trabalha basicamente com a geometria de linhas e objetos lineares no espaço 3D

- ✓ Pois a luz viaja em linha reta

Tony DeRose

desenvolveu um método de programação geométrica independente de coordenadas, o que simplifica o raciocínio geométrico.

Ou seja, ao invés de matrizes, utilizaremos operadores geométricos de alto nível (esses sim, são implementados com matrizes)

Mas antes,
um pouco de história...

Geometria
Afim

Geometrias

- geo = terra, metria = medida
- Agricultura, pirâmides, etc...

- **Euclides**: [sec 3 AC] formalizou a geometria a partir de alguns axiomas (sem sistema de coords).

- **Descartes**: [sec 17] introduz um sistema de coords, permitindo que conceitos geométricos fossem expressos matematicamente.

Geometria Afim

- Trata de “coisas” planas
  - ✓ Não há noção de distância, ângulo, ou orientação
- Elementos básicos
  - ✓ escalares (números)
  - ✓ pontos: usados para indicar uma posição
  - ✓ vetores livres: possuem direção e magnitude, mas nenhuma posição
    - ✓ vetor nulo ou zero
    - ✓ Livre?

Outras geometrias

- Sec 19: geometrias não Euclidianas [Lobachevski e Gauss], sugerem outros sistemas onde se aplicam outros axiomas.

- Nós veremos:
  - ✓ afim,
  - ✓ euclidiana e
  - ✓ projetiva

Pontos x Vetores

- Note que NÃO definimos um ponto especial, ou ORIGEM para o espaço afim.
  - ✓ Não há pontos especiais

- Por que distinguir pontos de vetores?
  - ✓ para tornar clara a intenção do programador
  - ✓ exemplo
    - ✓ escalar * vetor (OK)
    - ✓ escalar * ponto (?)

Convenções

- pontos: letras romanas maiúsculas
  - ✓ **P** e **Q**
- vetores: letras romanas minúsculas
  - ✓ em geral, sobrepostos com uma flecha
  - ✓ Mas vamos usar *t*, *u* e *v*
- escalares: em geral, letra grega
  - ✓ Mas vamos usar α e β

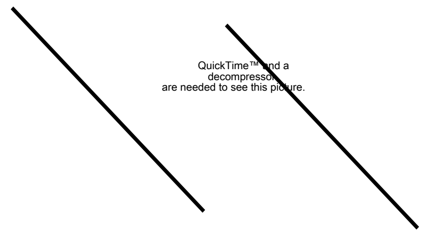
faz sentido somar pontos?

e multiplicar pontos por um escalar?

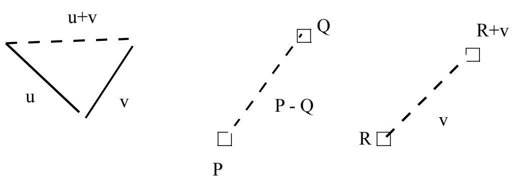
Combinações

- Multiplicação por escalar
  - ✓ $u = \alpha \, . \, v$, ou $u = v \, / \, \alpha$
- Adição de vetores
  - ✓ $t = u + v$ ou $t = u - v$
- Diferença entre pontos
  - ✓ v = **P** – **Q**
- Adição entre ponto e vetor
  - ✓ **Q** = **P** + v ou **P** = **Q** - v

Combinações afins



Operações afins



Soma de vetores

Subtração de pontos

Soma de ponto e vetor

Combinações afims

- Existe uma combinação particular entre pontos que consideraremos válida, denominada combinação afim.
  - ✓ Dado dois pontos **Q** e **P**, qual é o seu ponto médio **R**?
  - ✓ Mais genérico: qual o ponto **R** que divide **PQ** nas proporções α e 1-α? { sendo a entre [0,1]}
- Soluções
  - ✓ **R** = **P** + α(**Q** - **P**) (escala de **Q** - **P**)
  - ✓ **R** = (1 - α)**P** + ∂**Q** (média ponderada)

Combinação afim

- Dada uma sequência de pontos P1, P2, …, Pn, uma combinação afim é qualquer soma da forma α_1 P1 + α_2 P2 + … + α_n Pn

Onde a $\Sigma_i(\alpha_i) = 1$

Exercícios

- Dados 3 pontos tridimensionais não colineares, qual é a união de todas as suas combinações afins?
  - ✓ Resposta: todo o plano contendo os 3 pontos
- Qual a união de todas as suas combinações convexas?
  - ✓ Resposta: o triângulo (incluindo o seu interior) formado pelos 3 pontos.

Combinação convexa

É uma combinação afim
onde todo α
está no intervalo [0,1]

Geometria Euclidiana

Notação

- Vamos tomar a liberdade de escrever expressões do tipo

  (**P** + **Q**) / 2

  para indicar o ponto médio entre **P** e **Q.**

- Que é, claramente, ilegal!

A geometria afim não possui um mecanismo para tratar ângulos e distâncias.

A geometria euclidiana é uma extensão da geometria afim, que inclui o operador de produto interno (produto escalar)

O produto interno de dois vetores resulta em um escalar, e será denotado <*u.v*>.

Propriedades do produto escalar

- Positividade
  - ✓ <*u.u*> >= 0 e <*u.u*> == 0 iff *u* == 0;
- Simetria
  - ✓ <*u.v*> == <*v.u*>
- Bilinearidade
  - ✓ <*u*. (*v*+*w*)> == <*u.v*> + <*u.w*> e <*u*. α *v*> == α <*u.v*> onde α é um escalar

Para saber mais, veja um livro de álgebra linear

- **Ortogonalidade**
  - ✓ *u* e *v* são ortogonais (ou perpendiculares) se <*u.v*> == 0
- **Projeção ortogonal:** dados um vetor *u* e um vetor não nulo *v*, é conveniente decompor *u* como a soma de dois vetores $u = u_1 + u_2$, onde u_1 é paralelo a *v* e u_2 é ortogonal a *v*
  - ✓ u_1 = *v* <*u.v*> / <*v.v*> (pode-se ignorar <*v.v*> se |*v*| = 1)
  - ✓ $u_2 = u - u_1$
- verifique que u_2 é ortogonal a *v*.
- u_1 é chamado de projeção ortogonal de *u* sobre *v*.

Produto escalar

- **Produto escalar**
  - ✓ <*u.v*> = Σ (*u*i.*v*i)
- **Comprimento** de um vetor
  - ✓ |*v*| = sqrt[<*v.v*>]
- **Normalização**: dado um vetor não nulo *v*, a normalização de *v* resulta em um vetor de comprimento unitário, de mesma direção

  v' = *v* / |*v*|

Sistema de Coordenadas

QuickTime™ and a
decompressor
are needed to see this picture.

Produto Escalar

- **Distância entre pontos**
  - ✓ dist(**P**, **Q**) = |**P** - **Q**|
- **Ângulo** entre 2 vetores não nulos:
  - ✓ θ => ang(*u*,*v*) => $\cos^{-1}$(*u.v* / |*u*|.|*v*|) = $\cos^{-1}$(*u*' . *v*')
  - ✓ cos θ = <*u*'.*v*'>
  - ✓ Note que não sinal no ângulo entre *u*' e *v*', ou seja, se a direção do ângulo é horária ou anti-horária
  - ✓ Vamos voltar a falar sobre orientação quando o produto vetorial for introduzido

---

Sistema de coordenadas

- Sistema de coordenadas é o mecanismo utilizado para a definição dos objetos primitivos da geometria euclidiana (ponto, vetor, etc)
- Como representar pontos e vetores no espaço afim?
  - ✓ Álgebra linear: uso de 3 vetores linearmente independentes (i, j, k):

    v = ⋄⊠⊠i + ⋄⊠⊠j + ⋄⊠⊠k

Combinação linear e base

QuickTime™ and a
decompressor
are needed to see this picture.

Combinação linear

Base padrão

Coordenadas Homogêneas

- Para representar pontos e vetores em um espaço d-dimensional, utilizaremos vetores de comprimento d+1.
  - pontos tem a última coordenada = 1
  - vetores tem a última coordenada = 0
- Esse tipo de representação é denominado coordenadas homogêneas de um ponto ou vetor relativo a um frame F.

Recordação de álgebra linear

- Os vetores (i,j,k) formam uma base.
- A base mais comum (base padrão) utiliza vetores unitários ortogonais entre si: (1,0,0) (0,1,0) (0,0,1) chamada de base ortonormal.
- A tripla (⋄☒,⋄☒,⋄☒) define **a coordenada cartesiana** do vetor v.
- Porém, 3 vetores apenas não são suficientes para determinar a posição de um ponto.
- É necessário definirmos uma origem O para o sistema de coordenada.

Observações

- Primeiro, um axioma, para facilitar nossa forma de notação:
  0 * P = vetor nulo
  1 * P = P
- Atenção: "vetor" pode ser usado como
  - vetor livre: entidade geométrica
  - vetor de coordenada: forma de representação que pode ser usada para vetores livres e para pontos.

Frame de coordenadas

- Em um espaço afim, precisamos de uma forma de representar pontos além de vetores.
- Um frame de coordenadas em um espaço afim d-dimensional será definido por uma base ortonormal d-dimensional e um **ponto de origem**, de forma que um ponto P em 3D pode ser representado como:

P = ⋄☒☒i + ⋄☒☒j + ⋄☒☒k + O

Propriedades de coord. homogêneas

- A escolha de 1/0 para ponto/vetor não é arbitrária, ela possui algumas propriedades interessantes
  - v = P - Q: a última coordenada se cancela
  - seja U e V pontos ou vetores. Após várias operações da forma U-V, U+V ou ⋄U:
    - se a última coord = 0, o resultado é um vetor
    - se a última coord = 1, o resultado é um ponto
    - caso contrário, não é uma operação afim válida.
- Isso permite grande flexibilidade, como combinações do tipo:
  - centróide = (P+Q+R)/3

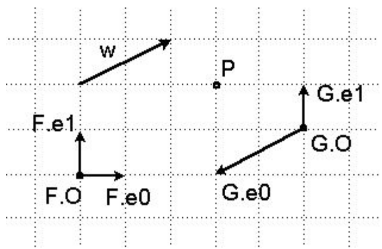
Sistemas de coord. alternativos

- Existem dois objetivos conflitantes em qualquer sistema de prog. geométrica:
  - pontos e vetores devem ser representados com respeito a algum sistema universal de coordenadas, que permita trabalhar com pontos e vetores simplesmente modificando suas coordenadas;
  - em geral é conveniente representar pontos de acordo com um sistema de coordenadas local
- Convenção:
  - sistema de coord. universal: frame padrão fixo, ortonormal: e0, e1, e2, O

Mudança de coordenadas: generalização

- Seja F o frame padrão
- Um frame arbitrário G pode então ser descrito com relação a F como:
  - Ge0 [F] = (g00, g01, 0)T
  - Ge1 [F] = (g10, g11, 0)T
  - GO [F] = (g20, g21, 1)T
- É dado um ponto P[F] = (⋄☒,⋄☒,⋄☒)T, ⋄☒=1, mas manteremos como incógnita para que as expressões funcionem para vetores livres também.
- Como determinar P[G] = (□0, □1 , □2)T?

Exemplo



w[F] =

P[F] =

w[G] =

P[G] =

Represente w e P nos frames de coordenadas F e G

Generalização

M = (G.e_0[F] I G.e_1[F] I G.O[F])

P[F] = M P[G]

Portanto: P[G] = M^{-1} P[F]

Observação importante: o OpenGL armazena matrizes em ordem de colunas (column-major order), enquanto C e C++ armazenam matrizes em ordem de linhas (row-major order)

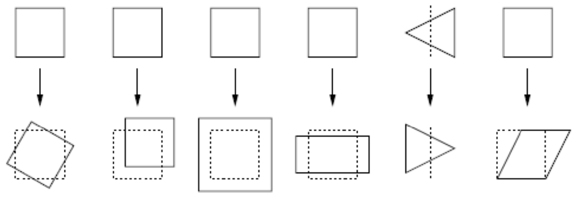
Exemplo de mudança de coordenadas

- Nosso exemplo mostrava que
  - P[F] = (3,2,1) e w[F]=(2,1,0)
  - como determinar suas coordenadas no frame G computacionalmente?
- Nosso objetivo é encontrar os escalares □0, □1 , □2 tal que P = □0G.e0 + □☒G.e1+ □2G.O
- Como F é um frame, os elementos de G podem ser descritos em função de F
  - G.e0 [F] =(-2,-1,0); G.e1 [F] = (0,1,0), G.O [F] = (5, 1, 1)
  - sistema linear com 3 equações e 3 incógnitas
  - solução (□0, □1 , □2) = (1,2,1)

Transformação Afim

- É uma classe de transformações que inclue:
  - rotação
  - translação
  - escala (uniforme e não uniforme)
  - reflexão
  - shearing
- Propriedades:
  - todas as transformações preservam combinações afins entre pontos
    - R = (1 - ⋄)P + ⋄Q => T(R) = (1-⋄)T(P) + ⋄T(Q)

Exemplos



Rotação Translação Escala Uniforme Escala não Uniforme Reflexão Cisalhamento

Transformações

- Reflexão
  - 2D: dada uma linha em um plano, a reflexão troca os pontos de lado simetricamente ao longo dessa linha.
  - 3D: dado um plano em 3D, a reflexão troca os pontos de lado simetricamente ao longo desse plano.
  - Pode ser considerado como um caso particular de escalonamento, em que o fator de escala é negativo
- Rotação
  - é definida para um ponto ou vetor, fixos no espaço.
  - casos básicos: rotação na origem, ao redor dos vetores bases, segundo a regra da mão direita.

Representação matricial

- As combinações afins são preservadas, portanto:
  - R = ⋄☒F.e0 + ⋄☒F.e1 + ⋄2F.O =>
  - T(R) = ⋄☒T(F.e0)+ ⋄☒T(F.e1)+ ⋄2T(F.O)
  - Assim, se R é um ponto ou vetor no frame F
  - e sabemos a transformação dos elementos do frame, então sabemos qual a transformação de R, ou
    - T(R)[F] = (T(F.e0)[F] | T(F.e1)[F] | T(F.O)[F]) (⋄☒☒⋄☒☒⋄☒☒
  - ou seja, aplicar uma transformação afim a um ponto/vetor equivale a multiplicar suas coordenadas por uma matriz, formada pela transformação dos elementos do frame.

Transformações

- Rotação ao redor de z
  - a origem e o vetor z não são alterados
  - o vetor unitário x é mapeado para (cos☒, sin☒, 0, 0)T
  - o vetor unitário y é mapeado para (-sin☒,cos☒, 0, 0)T
  - semelhante para os demais vetores de base
- Cizalhamento (Shearing)
  - pense em shearing como uma transformação que mapeia um cubo em um paralelogramo (fixe uma face do cubo, e desloque a face oposta, deformando as faces laterais).

Transformações

- Translação
  - translação por um vetor v mapeia qualquer ponto P para P + v.
  - vetores não são afetados por translação
- Mudança de escala
  - Mudança de escala uniforme é feita com relação a um ponto. Consideraremos a origem do frame padrão.
  - dado um escalar ☒, esta transformação mapeia um objeto (ponto ou vetor) de coordenadas (⋄☒,⋄☒ ,⋄☒ ,⋄☒)T para (☒⋄☒,☒⋄☒ ,☒⋄☒ ,⋄☒)T
  - para mudança não uniforme: (☒☒⋄☒,☒☒⋄☒ ,☒☒⋄☒ ,⋄☒)T

Transformações

- Shearing
  - xy-shear: o ponto P = (px, py, pz, 1)T é transladado pelo vetor pz(shx, shy, 0, 0)T, esse vetor é ortogonal ao eixo z e seu comprimento é proporcional a coordenada z do ponto P.
  - analogamente para xz e yz-shear.
- Composição de transformações
  - transformações afins são fechadas sob composição, ou seja, composições de transformações afins resulta em uma transformação afim.
    - T, S transformações: (T.S)(P) = (T(S(P))),
    - ou em notação matricial: MT MS P

Transformações

- Transformações complexas podem ser resolvidas por:
  - composições de transformações simples
  - determinação da imagem dos elementos da base.
    - em geral, esse método é mais simples
- Casos especiais:
  - Transformações rígidas ou Euclidianas: são transformações que preservam ângulos e comprimentos. Exemplo: translação, rotação e reflexão.
  - Transformações ortogonais ou homotéticas: preservam ângulos, mas não comprimentos. Exemplo: escalonamento uniforme.

Outros operadores geométricos

- Vimos até agora as operações:
  - mudança de sistemas de coordenadas
  - transformadas afins

- Outros operadores bastante utilizados:
  - produto vetorial
    - dado dois vetores, como determinar um terceiro vetor ortogonal aos dois primeiros?
  - orientação: dado 2 reais p e q, há 3 possibilidades de ordenação, $(p > q)$ ou $(p < q)$ ou $(p = q)$, o que permite a definição de um operador de orientação Or(p,q) que retorna -1, 0, ou 1 dependendo da ordem de p e q.

Produto Vetorial

- O produto vetorial é comumente definido no espaço 3D, pois ele se aplica apenas a vetores e não a pontos.
- Dado 2 vetores u, v, o produto vetorial é definido como:
  - u x v = (uyvz - uzvy, uzvx-uxvz, uxvy-uyvx)T
  - definição vale para um par de vetores no espaço 3D
- Determinante: u x v = (e | u | v) T
  - e : vetores da base

Propriedades do produto vetorial

- Simetria reversa: u x v = -(v x u)
  - u x u = 0
- Não associativo: ao contrário de outros produtos em álgebra, o produto vetorial não é associativo: (u x v) x w != u x (v x w)
- Bilinearidade:
  - u x (⋄v) = ⋄(u x v),
  - u x (v + w) = (u x v) + (u x w)

Outras propriedades do PV

- Perpendicularidade: (u x v) é perpendicular a u e a v (u e v linearmente independentes).
- Ângulo e área: |u x v| = |u| |v| sin ⌧
  - ângulo entre u e v: ⌧
  - em geral o PV não é usado para calcular ⌧ pois o produto escalar é mais simples de calcular.
  - |u x v| é igual a área do paralelograma cujos lados são dados por u e v.

Orientação

- Dados dois números reais p e q, uma função possível de orientação pode ser dado por: Or(p,q) = sign(q - p)
- Como estender esse conceito para dimensões superiores?
- Em um espaço d-dimensional, a orientação de (d+1) pontos pode ser definida pelo sinal do determinante da matriz composta pelas suas coordenadas homogêneas.

Interpretação

- Para um plano, a orientação de 3 pontos (P,Q,R) é +1 se o triângulo PQR é orientado no sentido anti-horário, -1 se orientado no sentido horário, e 0 se forem colineares.
- Em 3D, uma orientação positiva dos pontos PQRS significa que eles seguem uma espiral direita (mão direita), e zero se forem coplanares.
- Porque a coordenada homogênea primeiro?
  - caso contrário o sinal de determinante seria alternado para dimensões pares e impares

Intersecção de linhas

- Dado dois segmentos de linha PQ e RS em um plano, determinar se eles se intersectam.
  - possíveis problemas: linhas paralelas, junções, linhas colineares, etc.
- simplificação: apenas intersecções próprias (IP), com um único ponto em comum no interior dos segmentos.
    - se 3 pontos forem colineares, os segs não formam uma IP.
    - As linhas se cruzam iff P e Q se encontram em lados opostos de RS, e R e S se encontram em lados opostos de PQ
    - (Or(P,Q,R)*Or(P,Q,S)<0) && (Or(R,S,P)*Or(R,S,Q)<0)

O que você deve saber

- Programação geométrica
  - independente de coordenadas
- Geometrias
  - Afim
  - Euclidiana
- Frame de Coordenadas
- Coordenadas Homogêneas
- Transformações: translação, rotação, shear, escala, reflexão