

12/15/2013

FACULTATEA
DE
AUTOMATICA SI
CALCULATOARE

ELEMENTE DE GRAFICA PE CALCULATOR



Tema 4

„Sleeping Beauty”

Nota: tot ce nu este specificat in enunt este la latitudinea voastra. Daca aveti dubii, sunteti incurajati sa folositi forumul de la tema 4.

Scopul temei este sa implementati un sistem de animatie faciala ce reactioneaza la un sistem de particule. Pentru a realiza aceste cerinte veti primi o arhiva cu toate materialele artistice necesare.

Caracterul (tanara, „sleeping beauty”) din tema urmeaza la infinit aceeasi poveste: doarme, este trezita de un zgomot reprezentat ca niste note muzicale si devine surprinsa/sperziata, apoi cand realizeaza ca nu exista nici un pericol ci doar zgomot se enerveaza pentru ca a fost trezita degeaba. Cand zgomotul trece (reprezentat prin disparitia notelor muzicale) ea adoarme la loc. Cerinta poate fi observata cu usurinta in **videoul atasat** si in pozele din **anexe**.

Animatie Faciala:

Sistemul de animatie pe care trebuie sa il implementati se bazeaza pe un sistem de stari. Trecerea de la o stare la alta se face cursiv, prin interpolare (liniara/cubica/etc). Practic, daca timpul de trecere de la starea A la B este de N secunde atunci starea S de la K (<N) secunde este:

$$S = A * (1-t) + B * t, \text{ unde } t = K/N.$$

Daca avem o serie de stari, A -> B -> C, in cadrul in care trecem de la intervalul [A,B] la intervalul [B,C] putem inregistra o diferenta de timp fata de cadrul precedent care sa fie suficienta pentru trecerea la urmatorul interval dar nu exacta. In acest caz trebuie sa nu neglijam timpul ramas dupa trecerea in cel de-al doilea interval sau animatia nu va fi corecta. Pentru o iluminare corecta este necesara SI interpolarea normalelor.

Practic starile de animatie din tema sunt urmatoarele:

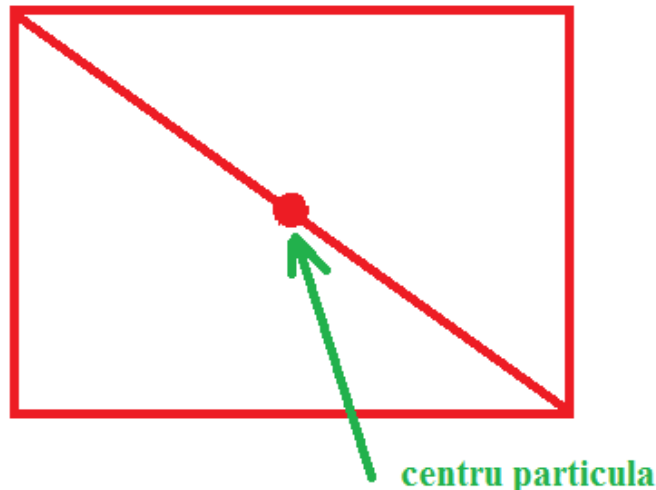
Somn -> surpriza -> enervare -> somn -> somn -> surpriza -> enervare -> somn -> somn -> surpriza -> enervare -> ...

Sistemul de animatie va fi executat de un vertex shader (pe GPU). CPU-ul doar va trimite starile(somn, surpriza, enervare) ca attribute sau ca texturi. In general pentru asemenea sisteme sunt recomandate automatele finite pentru ca faciliteaza controlul asupra procesului. Totusi stiind ca in cazul temei sunt doar 3 stari automatul finit este doar o sugestie. **Din materialele atasate veti folosi ca stari de animatie meshele „girl_annoyed.obj”, „girl_surprised.obj” si „girl_sleep.obj”. Pentru texturare caracter veti folosi „girl_texture.bmp”.**

Sisteme de Particule:

Un sistem de particule reprezinta vizual o multime de obiecte identice ce urmeaza un set comun de reguli. De exemplu un stol de pasari poate fi reprezentat ca un set de particule cum poate in acelasi timp fi reprezentata si o cascada.

Pentru a desena un sistem de particule trebuie sa creati un quad (2 triunghiuri), asemeni imaginii urmatoare:



Problema cu aceasta metoda este ca daca ne-am uita fix paralel cu planul particulei ar fi foarte greu sa observam particula in multe cazuri din cauza faptului ca proiectia ei ar avea o arie foarte mica pe ecran. Din aceasta cauza trebuie sa oferim garantia ca particula este **intotdeauna aliniata cu ecranul pentru ca proiectia ei sa aiba o arie constanta independent de unghiul de vedere. Acest concept se numeste billboards**. Pentru a putea sa implementam corect billboards trebuie ca in vertex shader sa calculam tot timpul centrul particulei, sa il trecem din model space in world space si apoi in view space. In viewspace putem sa calculam coordonatele de la colturi. De ce view space? Pentru ca ecranul este planul XoY la distanta -near in viewspace si stanga/dreapta reprezinta translatii pe oX iar sus/jos reprezinta translatii pe oY.

Scopul unui sistem de particule nu este sa simuleze una sau doua particule ci (sute de) mii! Mai mult desi particulele urmeaza acelasi set de reguli nu inseamna ca ele urmeaza exact aceleasi transformari. Imaginati-va 1000 de fulgi de zapada, fiecare este afectat de aceleasi forte dar traiectoriilor lor sunt diferite pentru ca forte au valori distincte pentru fiecare fulg (viteza, rotatie, dimensiune, etc).

Particulele din sistemul pe care trebuie sa il implementati urmeaza traiectoria unui proiectil. Din punct de vedere vizual particulele pleaca dintr-o parte a ecranului catre cealalta in mare majoritate ne-ocludand caracterul. **Din materialele atasate puteti folosi textura „music.bmp” pentru particule.**

Cerinte si barem orientativ:

- Sistem de animatie 85p (totul se face pe GPU!)
 - o Animatie de calitate 15p
 - o Trimitere corecta de date CPU -> GPU 10p
 - o Algoritmul este corect 30p
 - o Iluminare Phong in fragment shader (+normale calculate corect) 15p
 - o Texturare corecta 15p
- Sistemul de particule 55p (totul se face pe GPU!)
 - o Rezolvare ecuatiei sistem pe GPU 20p
 - o Billboards (pozitiile calculate in view space) 25p
 - o Texturare corecta cu alpha discard 10p
- Sincronizare intre animatie si sistemul de particule 10p

Bonusuri potentiale:

- Sisteme de particule complexe si realiste
- Sisteme de particule implementate cu geometry shader
- Aspect foarte! bun
- Coliziuni intre sistemul de particule si caracter
- Utilizare <http://www.makehuman.org/> (free ☺) sau <http://www.facegen.com/> (not free) pentru constructia propriului caracter cu numar mare de stari cu un lant de animatie complex (>10 stari)
- Subsurface Scattering pentru piele
- Orice creat de voi ce tine de grafica (computer science!) si nu este trivial.

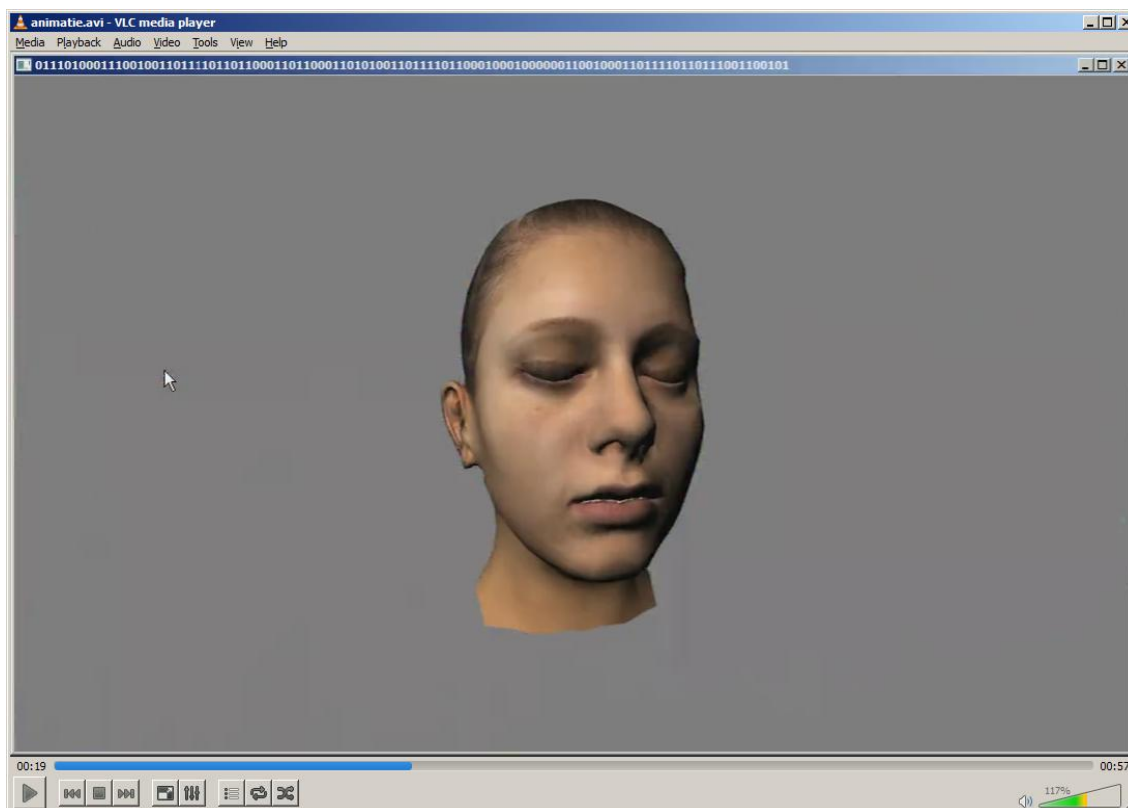
Responsabil: Lucian Petrescu

Termen: Duminica, 12 Ianuarie 2014, ora 23:55.

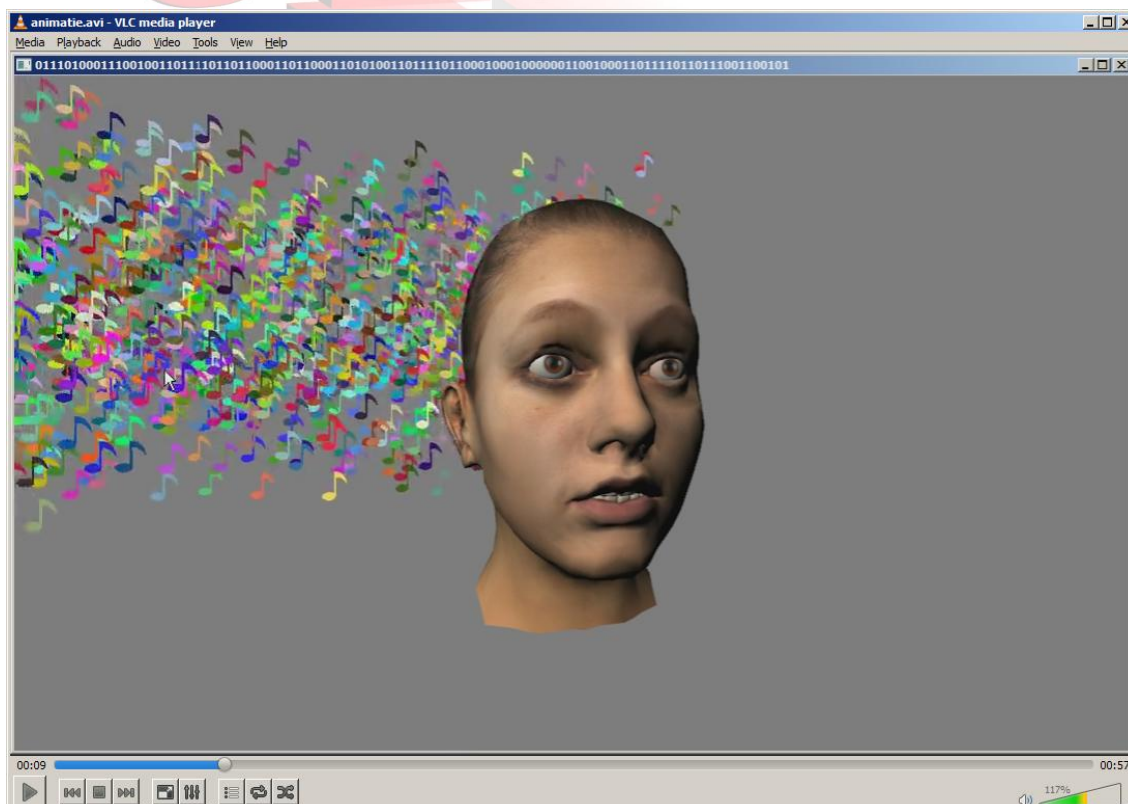
Obs: Tema este data pentru un timp de lucru de 2 saptamani (nu se considera ca si timp de lucru perioada vacantei).

Anexa:

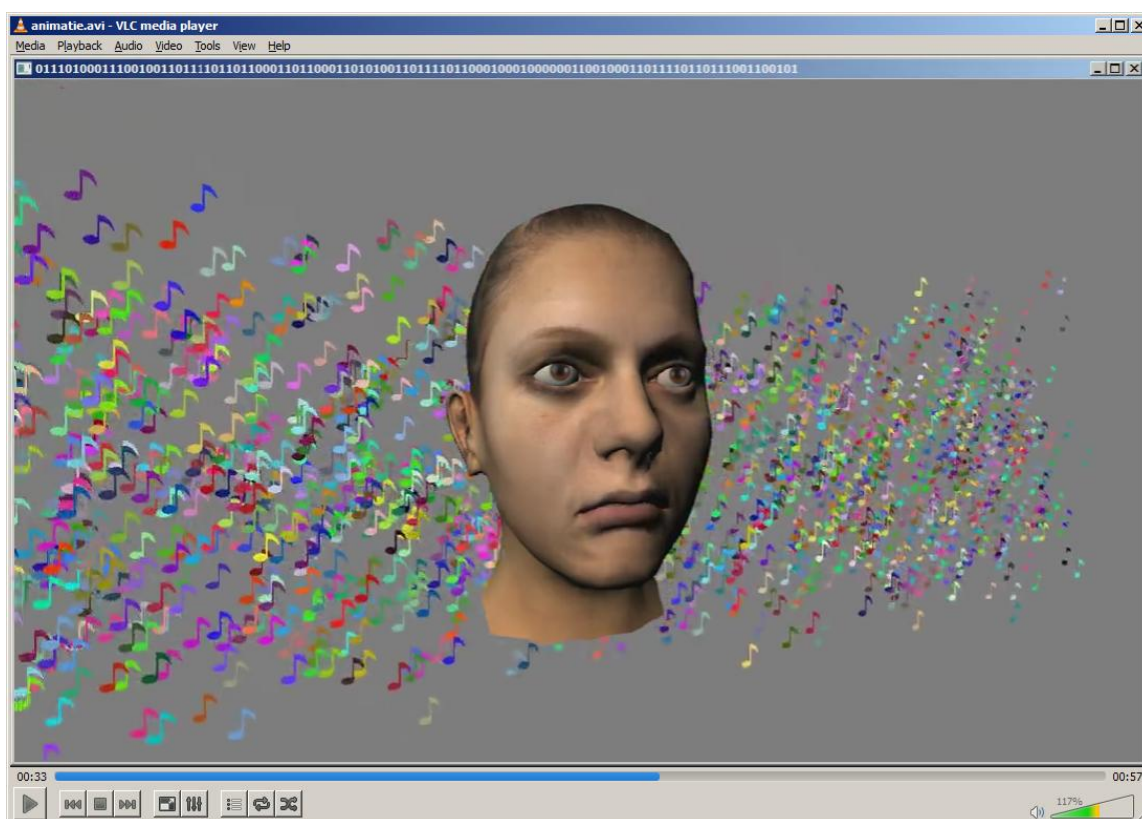
Caracterul adormit:



Caracterul surprins:



Caracterul suparat:



Caracterul intre 2 stari (adormit si surprins):

