

TOOL FOR FIREWORKS DESIGN AND SIMULATION

Kamil Floryán

Master Degree Programme (2), FIT BUT

E-mail: xflory02@stud.fit.vutbr.cz

Supervised by: Radek Bartoň

E-mail: ibarton@fit.vutbr.cz

Abstract: This thesis describes the design and implementation of particle system and a user interface of tool for fireworks design and simulation. The engine uses an XNA framework and an HLSL shading language. The thesis also compares applications focused on designing and simulation of fireworks. Applicability and demandings of applications for designing and simulation of fireworks among Czech and Slovak companies dealing professionally with firework are analysed as well.

Keywords: firework, pyrotechnics, simulation, XNA, HLSL, WPF, particle system

1 ÚVOD

Obsahem této práce je aplikace umožňující tvorbu a následnou simulaci ohňostrojevého představení. Aplikace je rozdělena na dva logické samostatné celky. První z nich je editor představení a druhou částí je následná vizualizace nakonfigurovaného představení.

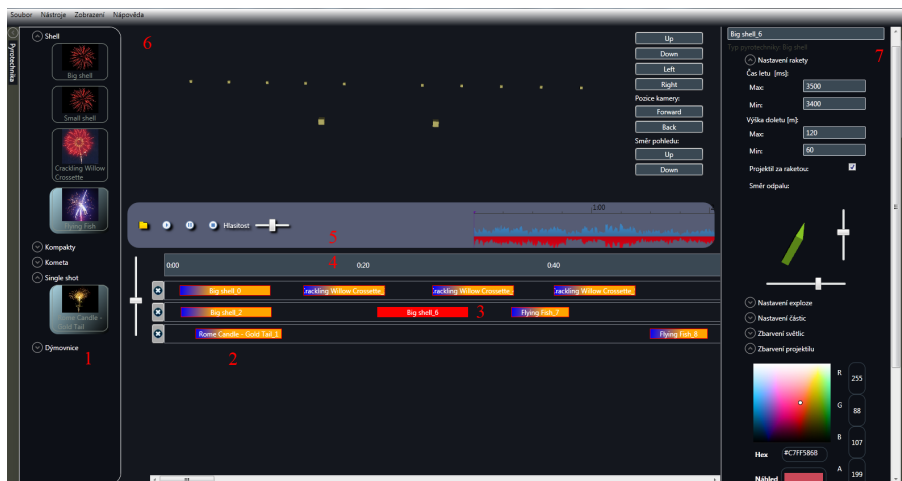
Primární cílovou skupinou, která by aplikaci mohla využívat jsou nadšenci pro zábavní pyrotechniku. Podle ohlasu na provedený průzkum, který se zabýval uplatněním aplikací simulujících pyrotechniku v praxi (a o kterém více pojednává kapitola 4) by si aplikace mohla nalézt uplatnění i při tvorbě profesionálních představení. Vizualizace sestaveného představení by byla vhodným prostředkem demonstrace názorné ukázky zákazníkovi. V případě, že by aplikace poskytovala export dat pro odpalovací systémy, mohla by aplikace umožňovat představení přímo sestavit a posléze i řídit.

2 POPIS PRODUKTU

Cílem mé práce bylo vytvořit editor ohňostrojevého představení, tj. aplikaci pro správu kompletního časování a pozicování jednotlivých odpalů pyrotechniky včetně detailního nastavení vlastností, které se podílejí na výsledném zobrazení efektu exploze. Následně takto nakonfigurované představení vizuálně co nejrealističtěji zobrazit a to v reálném čase.

Při návrhu uživatelského rozhraní a ovládání editoru byl kladen velký důraz na co nejintuitivnější umístění vybrané pyrotechniky do správného časového úseku a přiřazení správného odpalovacího systému. Pro tyto účely je využívána speciální grafická komponenta – časová osa. Přidávání pyrotechniky do představení je založeno na principu přetáhnutí (drag&drop) vybrané šablony pyrotechniky z předpřipravené kolekce na časovou osu. Každý objekt reprezentující pyrotechniku má (mimo jiné) pevně definovaný časový interval, po který se v představení vizuálně projevuje. To ovlivňuje velikost plochy, kterou tato pyrotechnika zaujme v rámci časové osy. Zároveň se v kolekci vložené pyrotechniky vytvoří nový unikátní druh pyrotechniky s vlastnostmi odvozených od použité šablony. Pro dosažení větší různorodosti představení a snížení nutného množství předdefinovaných šablon je možné vlastnosti každého vloženého kusu pyrotechniky modifikovat. Pro každou pyrotechniku lze modifikovat až 40 různých atributů rozdělených do několika kategorií.

Na obrázku 1 jsou popsány jednotlivé části editoru jejichž vysvětlení je uvedeno v následujícím výčtu:



Obrázek 1: Editor představení.

1. Dokovatelná a kategorizovaná kolekce šablon pyrotechniky. 2. Časové osy. 3. Vybraný kus pyrotechniky. 4. Časové značky náležící k časovým osám. Značky jsou odvozeny od předem definované délky představení (např. podle délky načteného hudebního podkladu). 5. Lišta pro manipulaci s hudebním podkladem. Vhodné k synchronizaci audia. 6. Rozložení odpalovacích systémů v třírozměrném prostoru. 7. Nastavitelné vlastnosti vybrané pyrotechniky – viz bod 3.

3 IMPLEMENTACE

Pro implementaci editoru byl zvolen framework *Windows Presentation Foundation 4.0* (WPF). Vizualizace využívá grafického frameworku *XNA 4.0*. Některé části kódu jsou přeneseny na GPU za pomoci shaderovacího jazyku *HLSL*.

3.1 EDITOR

Jak bylo již v kapitole 2 zmíněno aplikace využívá kolekci předpřipravených šablon pyrotechniky. Tyto šablony jsou načítány z lokálního úložiště v podobě XML souboru. Následně je provedena filtrace prostřednictvím technologie *LINQ* a přiřazení příslušných kolekcí do nabídky pyrotechniky.

Při výběru pyrotechniky se provede automatické navázání datového kontextu do zobrazení vlastností pyrotechniky v pravém panelu. K tomuto se využívá tzv. *Data Binding*, jenž obousměrně sváže data dané pyrotechniky s příslušným zobrazením (View).

3.2 VIZUALIZACE

Základním stavebním prvkem této práce je částicový systém. Podrobnější informace o částicových systémech lze dohledat na [2]. Ohňostroj může částicových systémů využívat hned několik a každý může fungovat na jiných principech. Pomocí částicového systému se vytváří vizuální efekty pyrotechniky. Implementace částicového systému zapouzdřuje kolekci částic, realizuje jejich vygenerování (funkčnost emitoru částic), jejich pohyb, vykreslení, aj. Různé druhy pyrotechniky se projevují různými efekty, které mohou být za pomoci hierarchické dědičnosti odvozeny od jiných – jednodušších efektů. Kontrolu nad částicovými systémy realizuje třída *ParticleSystemManager*. Částicový systém realizující světlice má pevný počet částic. Všechny částice typicky vznikají ve stejný okamžik a mají stejnou dobu života a v průběhu simulace se nové částice již negenerují. Proto jsou tyto částice uloženy v poli o velikosti počtu částic. Toto pole zaniká společně s částicovým systémem

při odstranění všech částic z částicového systému. *Částicový systém realizující projektil světlice* má předem daný pevný počet maximálně zobrazitelných částic. Částice vznikají i zanikají dynamicky a jsou ukládány do pole, které realizuje kruhovou frontu. Část obsahu tohoto pole je při vykreslení nového snímku nahrávána do vertex bufferu. Proto jsou v poli ukládány jednotlivé vrcholy částice a velikost pole je tedy čtyřnásobná k celkovému počtu částic.

Při vytváření záře u jednotlivých světlic či explozí je použit efekt *bloom*. Jedná se o rozzáření oblastí s vysokou hodnotou jasu. Tohoto efektu dosáhneme vykreslením daného částicového systému do textury. Tato textura je poté vertikálně a horizontálně rozmazána za pomoci efektu *blur*. Následně je původní obraz aditivně smíchán s rozmazaným obrazem což v důsledku vytváří efekt záře kolem vykreslených objektů. Při implementaci tohoto efektu bylo čerpáno z [1].

Pro urychlení výpočtů atributů popisujících částici (pozice, rychlost, rotace, průhlednost, aj.) jsou tyto výpočty částečně přesunuty na GPU a vypočteny rovněž pomocí shaderovacího jazyku HLSL.

4 VYHODNOCENÍ PRŮZKUMU

V průběhu vývoje aplikace, byla provedena analýza uplatnitelnosti a poptávky po aplikacích zabývajících se návrhem a simulací ohňostrojového představení. Několika českým a slovenským firmám pohybujících se v průmyslu zabývajícím se zábavní pyrotechnikou byl zaslán dotazník a video ukázky prvotního funkčního prototypu aplikace. Video ukázky především demonstrovaly manipulaci s editorem a výslednou vizualizaci. Výsledky průzkumu vyjadřuje tabulka 1.

Otázka	Celkem odpovědí	Ano [%]	Ne [%]
Již využívají podobný typ aplikace.	21	33,3	66,7
Domnívají se, že aplikace by pro ně mohla být užitečná.	18	88,9	11,1
Projevili zájem o používání výsledné aplikace.	18	83,3	16,7
Chtějí být informováni o výsledné podobě aplikace.	18	94,4	5,6
Předmět hodnocení	Celkem odpovědí	Průměrná známka	
Grafické rozhraní a ovládání editoru	10	1,75	
Vizualizace	13	1,54	

Tabulka 1: Vyhodnocení odpovědí průzkumu. Hodnoceno jako ve škole (1 – 5).

5 ZÁVĚR

V současné době je aplikace plně funkční a obsahuje celou řadu podporovaných druhů pyrotechniky a efektů. Za velmi přínosný lze označit i provedený průzkum, který mimo vyhodnocení otázek dotazníku poskytl velmi cennou zpětnou vazbu v podobě mnoha připomínek, návrhů a komentářů k předloženému konceptu aplikace.

V současnosti je na trhu několik aplikací zabývajících se návrhem a simulací ohňostrojových představení. Mezi nejznámější patří velmi komplexní aplikace *ShowSim* a *FINALE Fireworks*. Do budoucna je také plánována podpora pro dynamické přidávání nových druhů pyrotechniky prostřednictvím skriptů a podpora exportu dat přímo do odpalovacího zařízení.

REFERENCE

- [1] Wilhelmsen, P.: XNA Programming, digitalerr0r.wordpress.com/tutorials
- [2] Martin, A.: Particle Systems, web.cs.wpi.edu/~matt/courses/cs563/talks/psys.html