

UNIVERZA V NOVI GORICI
POSLOVNO-TEHNIŠKA FAKULTETA

**IZBOLJŠAVA ALGORITMA RAČUNALNIŠKE
ANIMACIJE ZA IGRALNE APARATE**

DIPLOMSKO DELO

Erik Cerkvénik

Mentor: prof. dr. Bogdan Filipič

Nova Gorica, 2013

ZAHVALA

Zahvaljujem se podjetju Gold Club, ki mi je omogočilo v diplomskem delu obravnavati izbrano problematiko, in prof. dr. Bogdanu Filipiču za mentorstvo pri izdelavi dela.

NASLOV

Izboljšava algoritma računalniške animacije za igralne aparate

IZVLEČEK

Računalniške animacije so pomembna sestavina interaktivnih elektronskih iger, ki jih razvija podjetje Gold Club. Način predvajanja animacij v igrah v začetnem razvojnem obdobju je zadoščal zahtevam trga in animatorjev, danes pa se pojavljajo težave s prezasedenostjo grafičnega pomnilnika in dolgimi časi nalaganja, posledično pa sta omejena število animacij in njihova pestrost. S to problematiko se srečujemo tudi pri igri slot. Cilj tega diplomskega dela je dosednji algoritem računalniške animacije izboljšati tako, da bo ustrezal zahtevam trga in animatorjev. Uvodoma analiziramo dosednji algoritem animacije z vidika zasedenosti pomnilnika in časa nalaganja, pregledamo možne rešitve in izberemo najprimernejšo, ki temelji na izrezovanju in pakiranju slik. Te izboljšave ovrednotimo najprej z izračuni, nato pa še z meritvami na igralnem aparatu. Rezultati potrjujejo, da izboljšani algoritem pomembno izboljšuje učinkovitost animacij, zato ga bomo vgradili v igro slot, v nadaljevanju pa še v ostale igre. S tem bo podjetje lahko trgu ponudilo igre s privlačnejšo in pestrejšo tematiko in krajšim časom nalaganja kot ostali proizvajalci.

KLJUČNE BESEDE

računalniška animacija, algoritem, elektronski igralni aparat, čas nalaganja, grafični pomnilnik, izrezovanje slik, pakiranje slik

TITLE

Improving computer animation algorithm for gaming machines

ABSTRACT

Computer animation is an important component of interactive electronic games developed by the Gold Club company. The way the animations in games were played in the early stage of development was sufficient to meet the demands of the market and the game animators. However, today there are difficulties with the graphical memory shortage and long loading times. Consequently, the number of animations and their variety are limited. This issue is encountered also in the slot game. The objective of this bachelor thesis is to improve the existing computer animation algorithm in a way that will meet the demands of the market and the game animators. Initially we analyze the existing animation algorithm regarding the memory requirements and loading times, review possible solutions and choose the most appropriate one based on image cutting out and packing. These improvements are evaluated first by calculation and then by benchmarking on the gaming machine. The results indicate that the improved animation algorithm significantly improves the animation efficiency. Therefore we are going to install it in the slot game first and subsequently also in other games. As a result, the company will be able to offer games with more varied themes and shorter loading times than other manufacturers.

KEYWORDS

computer animation, algorithm, electronic gaming machine, loading time, graphical memory, cutting out images, image packing

KAZALO

1	UVOD.....	1
2	PODJETJE IN NJEGOVI IZDELKI.....	3
2.1	O podjetju	3
2.2	Igralni aparati in ostali izdelki	3
2.2.1	Slot	4
2.2.2	Igre s kartami.....	6
2.2.3	Ruleta	7
2.2.4	Blagajna Vision.....	7
3	RAČUNALNIŠKE ANIMACIJE	9
3.1	Digitalne rastrske slike.....	9
3.2	Film.....	12
3.3	Večslojne animacije	12
3.4	3D animacija	13
4	PROBLEMATIKA DOSEDANJEGA ALGORITMA ANIMACIJE	15
4.1	Opis in slabosti algoritma	15
4.2	Izbira rešitve	15
5	IZBOLJŠAVA ALGORITMA	18
5.1	Zasnova.....	18
5.2	Izrezovanje koristne vsebine animacije	19
5.3	Odstranjevanje enakih objektov.....	22

5.4	Postavitev izrezanih objektov v eno sliko.....	23
6	OVREDNOTENJE IZBOLJŠANEGA ALGORITMA	28
6.1	Poraba grafičnega pomnilnika	28
6.2	Izračunani čas nalaganja slik	29
6.3	Izmerjeni čas nalaganja teme na igralnem aparatu	33
7	ZAKLJUČEK	35
8	LITERATURA	38

KAZALO SLIK

Slika 1: Poslovni prostori podjetja Gold Club	3
Slika 2: Primer mehanskega aparata slot	4
Slika 3: Ohišje igralnega aparata slot.....	5
Slika 4: Različne teme v igri slot	5
Slika 5: Dodatna igra v slotu: izbira žarnice vpliva na končni dobiček igre	6
Slika 6: Igra Texas hold'em poker: a) elektronska miza, b) igralčev zasebni zaslon z njegovimi kartami	7
Slika 7: Avtomatizirana ruleta: a) primer mize, b) prikaz na igralčevem zaslonu.....	7
Slika 8: Blagajna Vision: vmesnik za prenos denarja na igralno mesto	8
Slika 9: Digitalna rastrska slika: a) točke v sliki, b) točka ima svojo barvo oz. odtenek sivine.....	9
Slika 10: Osnovne računalniške barve: a) rdeča, zelena, modra, b) z mešanjem dobimo zeleno barvo	10
Slika 11: Točka barvne slike je sestavljena iz treh osnovnih barv: rdeče, zelene in modre.....	10
Slika 12: Ločljivost digitalne rastrske slike vpliva na razpoznavnost vsebine	11
Slika 13: Prosojnost slike: a) izvorna slika, b) alfa kanal, c) končna slika	11
Slika 14: Film je sestavljen iz več zaporednih slik	12
Slika 15: Komponente večslojne animacije	13
Slika 16: Model 3D animacije: upodobljena kitara in njen žični model.....	14
Slika 17: Ključni položaji hodeče osebe	14
Slika 18: Slika animacije z veliko praznine okrog objektov	18

Slika 19: Ideja rešitve: slika animacije le s potrebnimi objekti	18
Slika 20: Vse slike animacije premikajočega se objekta	19
Slika 21: Primer slike animacije premikajočega se objekta.....	19
Slika 22: Detekcija vsebine po oseh x in y	19
Slika 23: Izrezan objekt.....	20
Slika 24: Izrezani objekti celotne animacije	20
Slika 25: Enostavno izrezovanje v primeru zelo razpršenih objektov po sliki	21
Slika 26: Primer izreza objektov s škarjami.....	21
Slika 27: Enostavno izrezovanje slike z objektoma v nasprotnih kotih: a) izvorna slika, b) rezultat enostavnega izreza.....	21
Slika 28: Pasovi praznin in izbrani rez.....	22
Slika 29: Več objektov, več praznih pasov	22
Slika 30: Rezultat delitve slike in ponovnega izreza	22
Slika 31: Izrezani objekti	22
Slika 32: Končni rezultat izrezovanja objektov z odstranjenimi dvojniki	23
Slika 33: Osnovna zamisel postavljanja objektov v sliko	24
Slika 34: Stopničavost pri postavitvi več objektov v sliko	25
Slika 35: Miselni proces med postavljanjem kock.....	26
Slika 36: Postopek postavljanja pravokotnih objektov na površino	26
Slika 37: Rezultat postavljanja pravokotnih objektov na površino po različnih kriterijih razvrščanja: a) po površini, b) po širini.....	27

KAZALO TABEL

Tabela 1: Prostorske izboljšave po izrezovanju slik premikajočega se objekta.....	28
Tabela 2: Zasedenost predpomnilnika z grafičnimi temami pred izrezovanjem slik in prihranek po njem	29
Tabela 3: Izračunani dostopni čas do slik pred izboljšavo in po njej.....	31
Tabela 4: Izračunani čas branja vseh slik pred izboljšavo in po njej	31
Tabela 5: Izračunani skupni čas nalaganja slik pred izboljšavo in po njej	32
Tabela 6: Izmerjeni časi nalaganja tem v igri slot pred izboljšavo in po njej	34

1 UVOD

Z razvojem računalniške tehnologije in razširjenostjo cenovno dostopne zmogljive strojne opreme so se tudi interaktivne računalniške aplikacije izboljšale tako po vsebinski pestrosti, vizualnih učinkih in animacijah kot tudi po končni uporabnikovi izkušnji. Enako velja na področju elektronskih igralnih aparatov v kazinojih, zato tehnično zastarele, monotone in neprivlačne igralne aparate zamenjujejo s sodobnejšimi.

Tak primer je tudi interaktivna igra slot podjetja Gold Club, ki jo obravnavamo v tem diplomskem delu. Ta igra vsebuje računalniške animacije, ki pa so se v dosedanji izvedbi izkazale za neustrezne v pogledu uporabe računalniških virov: postale so preobsežne za grafični pomnilnik in čas nalaganja njihovih slik v pomnilnik je bil predolg. Zaradi teh slabosti dosedanji način predvajanja animacij ni več zadovoljeval potreb trga in zahtev animatorjev. Da bi te težave odpravili, smo algoritem računalniške animacije za igro slot izboljšali.

Cilji dela so bili proučiti različne vrste računalniških animacij in njihove značilnosti, analizirati slabosti dosedanjega algoritma računalniške animacije v igri slot in nato poiskati primerno rešitev oz. izboljšavo. Izbrano rešitev smo želeli tudi implementirati in ovrednotiti, tako da bi bila primerna za vgradnjo v igralne aparate.

Uporabljena metodologija je temeljila na poznavanju računalniških animacij, igre slot in izkušnjah v podjetju. Najprej smo analizirali slabosti dosedanjega algoritma animacije z vidika zasedenosti grafičnega pomnilnika in časa nalaganja, nato pa še uporabljene animacije v igri. Glede na ugotovitve in smiselnost izvedbe smo izbrali primerno rešitev. Rešitev oz. izboljššan algoritem smo implementirali v programskem jeziku C# za platformo Microsoft.NET v operacijskem sistemu Windows. Izboljšave smo najprej ovrednotili z izračuni, nato pa še z meritvami na igralnem aparatu. V zadnjem koraku smo prenesli izboljššan algoritem v redno proizvodnjo igralnega aparata slot.

Vsebinsko je diplomsko delo razdeljeno na sedem poglavij. Uvodu sledi drugo poglavje, v katerem opisujemo podjetje Gold Club in nekaj značilnih igralnih aparatov, ki jih izdelujejo v podjetju. V tretjem poglavju uvodoma predstavimo

osnove računalniške animacije, od predstavitve digitalnih slik do različnih vrst animacij. Slabosti dosedanjega algoritma, možne rešitve in utemeljitev izbire rešitve obravnavamo v četrtem poglavju, postopen razvoj rešitve pa opišemo v petem poglavju. V šestem poglavju z izračuni in meritvami ovrednotimo izboljšani algoritem animacije z vidika porabe grafičnega pomnilnika in časa nalaganja. V zaključnem, sedmem poglavju povzamemo vsebino dela in rezultate ter predlagamo nadaljnje korake.

2 PODJETJE IN NJEGOVI IZDELKI

2.1 O podjetju

Podjetje Gold Club, z osnovnim kapitalom 208.000 EUR, je srednje veliko podjetje ustanovljeno leta 1998 in je danes svetovno uveljavljen proizvajalec igralnih aparatov za kazinoje. Sedež in poslovni obrat ima v Sežani, kjer zaposluje 90 ljudi (slika 1).



Slika 1: Poslovni prostori podjetja Gold Club

Najstarejši in najbolj znan igralni aparat, ki ga izdelujejo v podjetju, je avtomatizirana elektromehanska ruleta, s katero si je podjetje že kmalu po ustanovitvi utiralo pot na mednarodne trge. Ti so zakonsko urejeni in zahtevajo ustrezna dovoljenja in certifikate, saj morajo biti igralni aparati izdelani v skladu s strogo zakonodajo in pravili delovanja. Danes podjetje nastopa že po vsem svetu, predvsem v Evropi, Aziji in Južni Ameriki. Poleg avtomatizirane rulete proizvaja tudi druge vrste interaktivnih elektronskih iger in strežniških sistemov: namizne igre s kartami, kot so npr. Blackjack, Baccarat in Texas hold'em poker, igralne aparate slot z različnimi tematikami, blagajne in sisteme za nadzor lastnih igralnih aparatov. Ima lastne oddelke za marketing, proizvodnjo, raziskave in razvoj in testiranje ter servis, ki je strankam neprekinjeno na razpolago za tehnično pomoč (Gold Club, 2012).

2.2 Igralni aparati in ostali izdelki

Igralni aparati podjetja Gold Club so elektronske ali elektromehanske različice klasičnih kazino iger. So moderno oblikovani, enostavne konstrukcije in vsebujejo

sodoben interaktivni grafični vmesnik z zasloni na dotik, kar še poenostavi njihovo uporabo. Zahtevna programska oprema v ozadju omogoča povezljivost aparatov na različne informacijske sisteme v kazinoju in vsebuje varnostne mehanizme proti nepooblaščenim dostopom, manipulacijam in drugim zlorabam. Aparati so izdelani v skladu s strogimi igralniškimi zahtevami o varnosti in delovanju (certifikati).

Ostali izdelki podjetja so strežniške aplikacije za manjše igralnice, kot npr. blagajna za izdajo vračljivih elektronskih igralnih kartic, sistem za spremljanje finančnega prometa in igranosti aparatov.

V nadaljevanju predstavljamo najpomembnejše izdelke podjetja. Slikovno gradivo o njih je povzeto po (Gold Club, 2013).

2.2.1 Slot

Igralni aparat slot, ki ga proizvaja podjetje Gold Club, je modernejša različica klasičnega elektromehanskega aparata, ki je prikazan na sliki 2. Klasični aparat ima na desni strani gibljivo ročico, ki jo igralec potegne navzdol in s tem zavrti tri ali več cilindrov (ang. reel) v srednjem delu aparata. Podobno kot pri kolesu sreče igralec počaka na umiritev cilindrov, glede na dobljeno kombinacijo simbolov (na sliki 2 vidimo simbole »7«) pa lahko zadene dobiček, ki je prikazan v tabeli dobitnih kombinacij v zgornjem delu aparata.



Slika 2: Primer mehanskega aparata slot

Elektronska različica aparata podjetja Gold Club nima ročice, mehanske cilindre pa nadomešča računalniški zaslon z interaktivnim vmesnikom. Po zunanosti je podobna klasični igri, ponuja pa tudi različice ohišij, bodisi z zasloni na dotik ali brez njih kot tudi z dodatnim zaslonom zgoraj za dinamični prikaz različnih informacij in plačilnih tabel dobitnih kombinacij (slika 3).



Slika 3: Ohišje igralnega aparata slot

Za igralce je pomemben del igre téma oz. kontekst simbolov, nekaj jih je prikazanih na sliki 4. Zamenjava teme pri klasičnih aparatih ni bila enostavna, saj je bilo potrebno zamenjati celoten igralni del aparata, ker ni omogočal tovrstne zamenjave. Elektronska različica aparata zamenjavo teme bistveno poenostavi – zamenjati je potrebno le programsko opremo in pripadajoče elemente, kot so tematske slike na ohišju.



Slika 4: Različne teme v igri slot

Prednost elektronskega aparata slot je tudi dodatna igra v igri (bonus igra, primer na sliki 5) ali celo možnost izbire grafične teme po igralčevi želji na aparatu z več temami.



Slika 5: Dodatna igra v slotu: izbira žarnice vpliva na končni dobiček igre

2.2.2 Igre s kartami

Podjetje izdeluje tudi nekaj aparatov za igre s kartami, kot so npr. namizni Blackjack, Baccarat in Texas hold'em poker, ter nekaj različic video pokra v ohišju aparata za igro slot.

Aparata za Blackjack in Baccarat sta po zunanjem videzu podobna klasični polkrožni mizi, igri pa imata podobna pravila. V igri lahko sodeluje največ pet igralcev, na mestu krupjeja pa stoji večji zaslon. Na njem se prikazujejo karte na mizi, pomembnejše informacije o igri in navidezni krupje. Ta nakazuje potek igre, deli karte, nagovarja igralce, objavlja rezultate oz. vodi igro, tako kot se to počne v realnosti. Vsak igralec ima pred seboj zaslon na dotik, ki prikazuje njegovo finančno stanje, zasebne in krupjejeve karte, preko zaslona pa igralec tudi igra. V igri Baccarat se igralcu na zaslonu prikazujejo tudi nepogrešljive informacije o preteklih igrah, to so rezultati partij v posebnem tabelaričnem zapisu. To nadomešča ročno beleženje teh informacij in s tem olajša igranje.

Enako kot aparata za Blackjack in Baccarat je tudi aparat za igro Texas hold'em poker po videzu podoben klasični mizi, kot jo vidimo na sliki 6a. Na osrednjem prikazovalniku se prikazuje stanje na mizi (zastavljeni žetoni, skupne karte, vloga igralca in ostale javne informacije), vsak igralec pa igra preko svojega zaslona na dotik (slika 6b).



a)



b)

Slika 6: Igra Texas hold'em poker: a) elektronska miza, b) igralčev zasebni zaslon z njegovimi kartami

2.2.3 Ruleta

Elektromehanska avtomatizirana ruleta je najstarejši in hkrati najbolj izpopolnjen aparat oz. igra podjetja Gold Club. Po pravilih je igra enaka klasični, vendar je brez krupjeja, saj aparat samodejno zavrti kolo rulete, požene žogico, prebere padlo številko na kolesu, obračuna stave in dobitke ter se pripravi na naslednjo igro. Rulete so različnih oblik in imajo različno število igralnih mest. Značilen primer mize avtomatizirane rulete vidimo na sliki 7a, na sliki 7b pa prikaz na igralčevem zaslonu.



a)



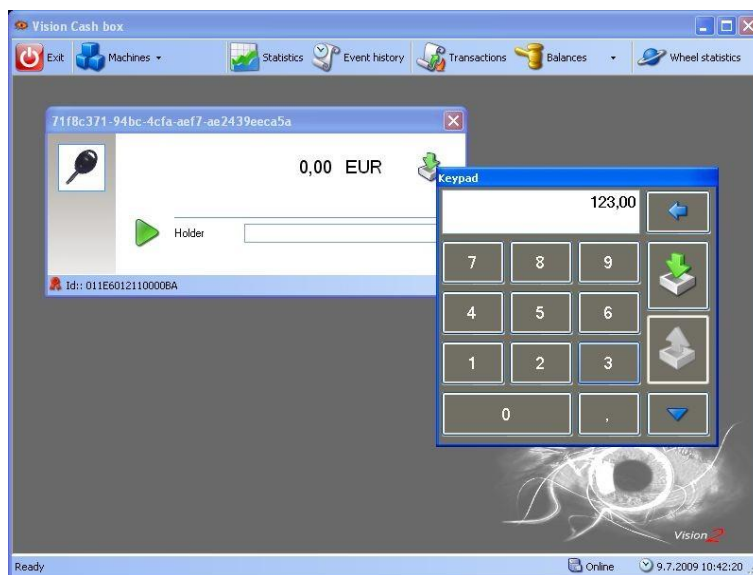
b)

Slika 7: Avtomatizirana ruleta: a) primer mize, b) prikaz na igralčevem zaslonu

2.2.4 Blagajna Vision

Poleg igralnih aparatov ponuja podjetje tudi elektronsko blagajno za uporabo z lastnimi aparati, namenjeno predvsem manjšim igralnicam, ki nimajo celotnega informacijskega sistema. Preko vmesnika (slika 8) lahko zaposleni nakazujejo denar na igralna mesta, izplačujejo dobitke, lastniki pa imajo zaradi nadzora na voljo vpogled v stanje aparata, vplačilne in izplačilne transakcije, razne dogodke ipd. Za

zahtevnejšim uporabnikom je na voljo tudi spletni dostop do podatkov igralnih aparatov v igralnici preko programa Vision WebReporter. Ta omogoča tudi mesečne izpise o prometu, spremljanje servisnih dostopov do notranjosti aparata in vpogled v trenutno finančno stanje.



Slika 8: Blagajna Vision: vmesnik za prenos denarja na igralno mesto

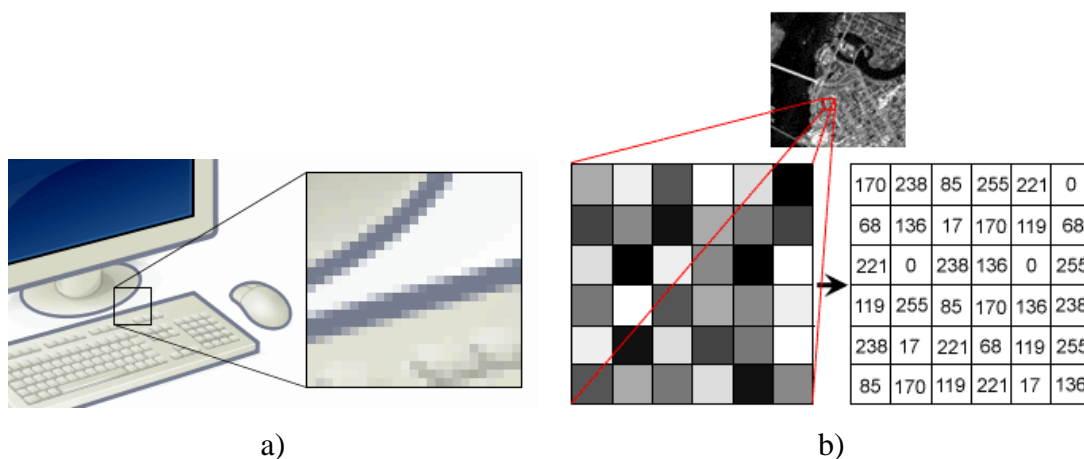
3 RAČUNALNIŠKE ANIMACIJE

Računalniške animacije razumemo predvsem kot vizualni učinek grafičnih objektov na zaslonu, npr. premikajoče se karte pri igri Blackjack in vrteče cilindre v igri slot, ali pa kot enostaven prikaz animacije, kot je predvajanje filma. Pristopov za doseg učinka animacije je več, odvisno od kompleksnosti in števila objektov v animaciji.

3.1 Digitalne rastrske slike

Med uporabo računalnika gledamo v zaslon, na katerem se prikazujejo različne vsebine, kot so npr. spletne strani. To je v osnovi celozaslonska digitalna slika, ki jo računalnik nenehno izrisuje. Ker je ta osnova za prikazovanje vsebine, bomo najprej opisali, kako je digitalna računalniška slika sestavljena.

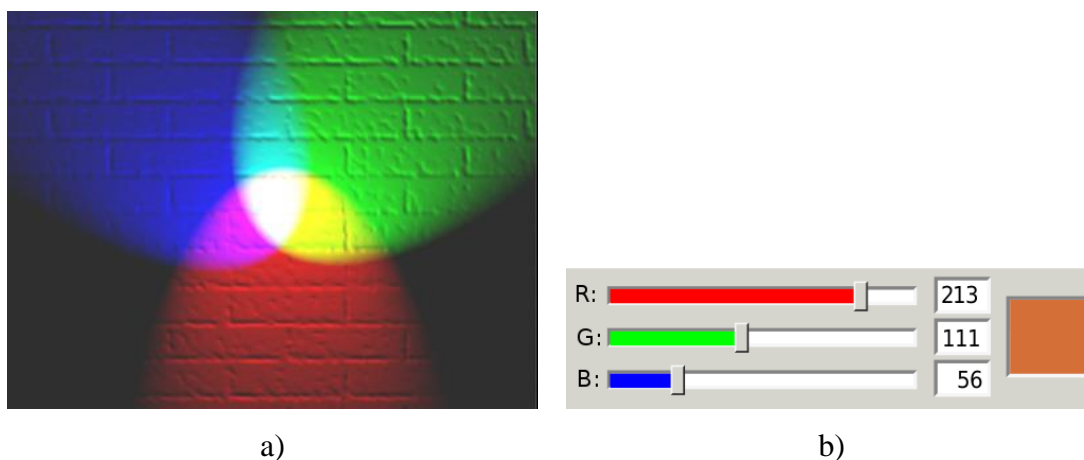
Računalniška rastrska slika (Raster graphics, 2011) je sestavljena iz mnogo majhnih točk, imenovanih tudi slikovni elementi ali piksli (angl. pixel), prikazanih v povečavi na sliki 9a. Vsaka točka ima svojo barvo, oziroma, na primeru slike 9b, odtenek sivine.



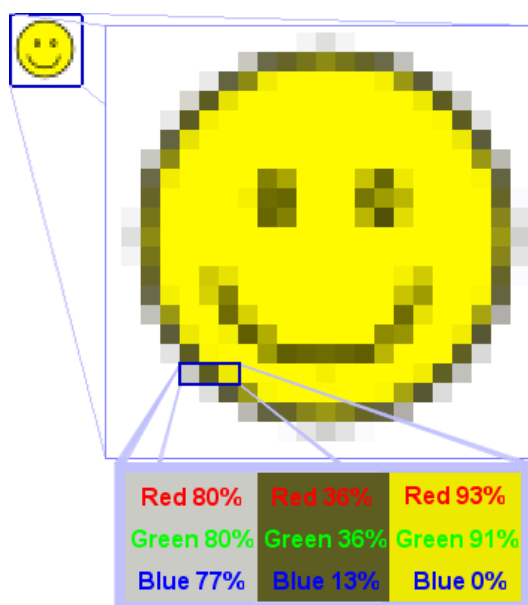
Slika 9: Digitalna rastrska slika: a) točke v sliki, b) točka ima svojo barvo oz. odtenek sivine

Točka enobarvne slike je tipično črna ali bela, točka sivinske slike je v nekem odtenku ene osnovne barve, točka barvne slike pa je v barvnem odtenku, sestavljenem iz treh osnovnih barv: rdeče, zelene in modre; znane kot RGB (angl. red, green, blue) barve, prikazane na sliki 10a (RGB color model, 2011), na sliki 10b

pa je prikazan koncept mešanja RGB barv. Slika 11 dodatno ilustrira sestavo barve v točki barvne slike.



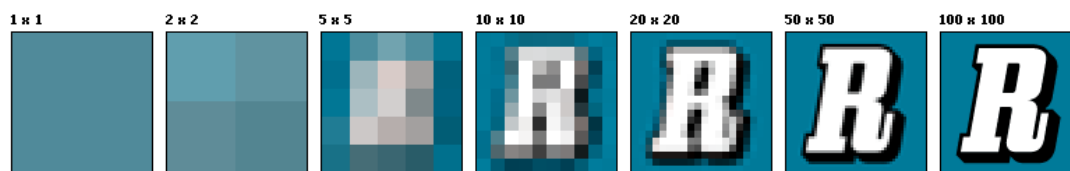
Slika 10: Osnovne računalniške barve: a) rdeča, zelena, modra, b) z mešanjem dobimo želeno barvo



Slika 11: Točka barvne slike je sestavljena iz treh osnovnih barv: rdeče, zelene in modre

Ker je digitalna slika sestavljena iz točk, njihovo število vpliva na razpoznavnost vsebine. Ločljivost se meri v številu točk v horizontalni smeri (os x) in vertikalni smeri (os y). Slika z današnjega običajnega kompaktnega digitalnega fotoaparata ima npr. ločljivost približno 4.000×3.000 točk (širina \times višina), kar je skupno kar 12.000.000 točk.

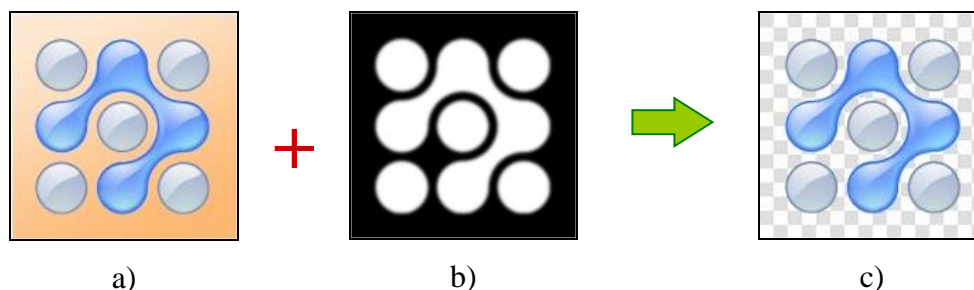
Na sliki 12 je nekaj primerov različne ločljivosti slike, ki prikazuje črko R. Pri ločljivosti 5×5 točk vidimo nejasen zmazek, pri ločljivosti 20×20 točk je črka R že lepo prepoznavna, najboljši videz pa je pri najvišji dani ločljivosti 100×100 točk.



Slika 12: Ločljivost digitalne rastrske slike vpliva na razpoznavnost vsebine

Večje kot je število točk v sliki, več računalniškega pomnilnika ali medija ta slika zaseda. Prej omenjena barvna slika s fotoaparata z 12.000.000 točkami v nestisnjeni (surovi) obliki zaseda približno 34 MB pomnilniškega prostora, kar pomeni le 20 slik na zgoščenko (CD) kapacitete 700 MB. Vsaka točka namreč zasede tri bajte, en bajt na barvo. Vsak bajt predstavlja enega izmed 256 odtenkov osnovne barve iz slike 10b. Ker je shranjevanje nestisnjenih slik v večini primerov nepraktično, jih shranjujemo v stisnjeni obliki, tako da zasedajo bistveno manj pomnilniškega prostora (npr. le 10 % v primerjavi z nestisnjeno obliko), trpi pa lahko razpoznavnost vsebine (Lossy compression, 2012).

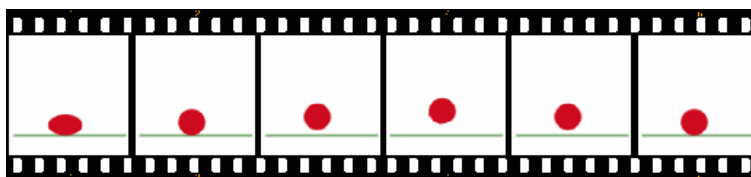
Digitalne slike so lahko tudi prosojne, v kolikor poleg barv izvirne slike vsebujejo tudi alfa kanal (slika 13a, b). Ta je kot posebna »četrt« barva v sliki poleg rdeče, modre in zelene in podaja informacijo o prosojnosti posamezne točke (Alpha compositing, 2012); prosojnost točke je odvisna od svetlosti »četrt« barve. Na sliki 13b črna barva predstavlja popolno prosojnost, bela pa polno barvo (točke brez prosojnosti) s končnim učinkom, kot ga vidimo na sliki 13c. Odtenki sivine med črno in belo barvo predstavljajo delno prosojnost.



Slika 13: Prosojnost slike: a) izvorna slika, b) alfa kanal, c) končna slika

3.2 Film

Človeku najbolj poznana in razumljiva animacija je film. Sestavljen je iz mnogo zaporednih slik, ki se med predvajanjem tako hitro menjujejo, da gledalcu zamenjave niso moteče, dajejo pa učinek gibanja objektov na zaslonu. Na sliki 14 vidimo odsek filma, ki vsebuje šest zaporednih slik. Na vsaki sliki se objekt (rdeča žoga) nahaja drugje, kar ob dovolj hitri menjavi slik na zaslonu daje učinek odbijajoče se žoge oz. animacijo.

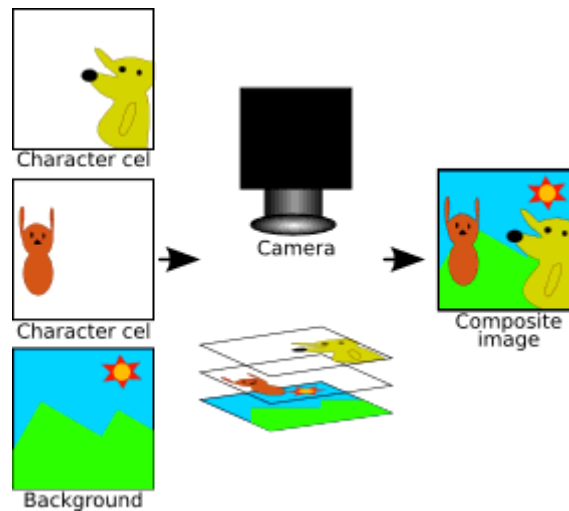


Slika 14: Film je sestavljen iz več zaporednih slik

Tipičen celovečerni film dolžine 90 minut, v katerem se zvrsti 25 slik na sekundo, obsega zajetnih 135.000 slik. V televizijskem formatu PAL (PAL, 2012) zasedejo vse slike filma v nestisnjeni obliki kar 214 zgoščenk. Ravno tako kot shranjujemo slike zaradi praktičnih razlogov v stisnjeni obliki, to počnemo tudi s filmom. V tem primeru bi omenjeni celovečerec lahko zasedel le eno zgoščenko.

3.3 Večslojne animacije

Ker so računalniške animacije po vzoru filma prostorsko zahtevne, so za nekatere vsebinsko enostavnejše primere ustrežnejši drugačni pristopi. Eden od načinov je uporaba enostavne večslojne tehnike, kjer na vsakem sloju predstavimo po en osnovni objekt. Končna slika (angl. composite image) na sliki 15, ki jo vidimo na zaslonu, je sestavljena iz več slojev: ozadja (angl. background) ter dveh likov (angl. character cell), postavljenih eden nad drugim v želenem zaporedju.



Slika 15: Komponente večslojne animacije

Vse sloje (slike) vključno z ozadjem lahko med potekom animacije premikamo, zamenjamo, obračamo ipd. Slojev je lahko več in ni nujno, da so vsi vidni v istem trenutku. Prednost takega pristopa je, da zaseda precej manj računalniškega pomnilnika v nasprotju s filmom, ker se hranijo le osnovni sloji (slike) ter podatki o njihovih premikih med animacijo. Ravno zaradi nizke prostorske zahtevnosti se podoben pristop uporablja tudi na multimedijskih spletnih straneh, npr. s tehniko Adobe Flash (Kerman, 2002). Slabost tega pristopa pa je ravno v omejenem številu osnovnih slojev, zaradi česar trpi pestrost animacije.

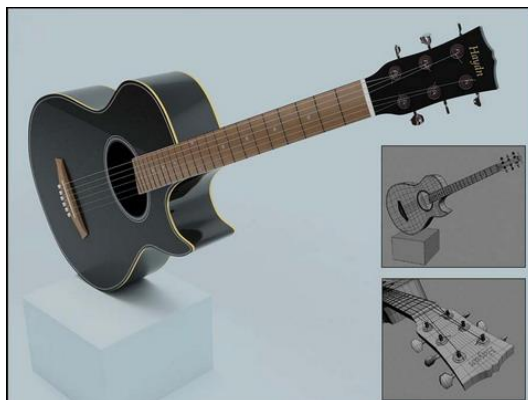
3.4 3D animacija

Ta vrsta animacije je zelo razširjena in je danes prisotna skoraj v vseh računalniških igrah. V okviru takih animacij se skrivajo modeli predmetov, npr. stol, miza, osebe, ki jih računalnik upodobi v realnem času in prikaže na zaslon kot npr. kitaro na sliki 16 (Erzetič in Gabrijelčič, 2010).

Prednost 3D animacije je spremenljivost objektov glede na odzive uporabnika ali stanje igre. V primeru filma ali večslojne animacije je nemogoče prikazati kitaro iz poljubnega zornega kota ali različnih barv, prav tako je nemogoče dinamično animirati strune, med tem ko igralec »brenka« s klikanjem po njih.

Slabost 3D animacije je zahtevnost programske opreme, saj je potrebno modele najprej narisati v ustreznem grafičnem programu, jih nato uporabiti in spreminjati

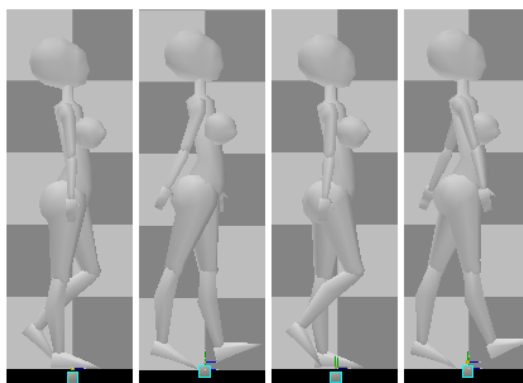
njihove lastnosti v procesu igre (barvo, zorni kot, premikanje strun itd.) ter jih z ustreznim grafičnim pogonom upodobiti uporabniku na zaslon (Patmore, 2003).



Slika 16: Model 3D animacije: upodobljena kitara in njen žični model

Definicija animacije objekta največkrat vsebuje le ključne vrednosti spremenljivih lastnosti, npr: začetna barva, eno ali več vmesnih barv, končna barva. Preostale barve, ki niso eksplicitno definirane, grafični pogon samodejno izračunava. Na tak način dosežemo barvne prehode le z definiranjem nekaj ključnih barv, ne pa vseh barv v vsakem trenutku animacije.

Kot primer vzemimo prikaz hoje. Animator vnaprej pripravi nekaj ključnih položajev delov telesa, npr. rok, nog, trupa, glave ipd., ki jih animacijski algoritem ustrezno uporabi za izračun vmesnih položajev in tako ustvari animacijo hodeče osebe (slika 17).



Slika 17: Ključni položaji hodeče osebe

4 PROBLEMATIKA DOSEDANJEGA ALGORITMA ANIMACIJE

4.1 Opis in slabosti algoritma

Dosedanji algoritem prikazovanja animacij v elektronskih igrah podjetja deluje podobno kot predvajanje filma in potrebuje vse slike animacije naložene v grafičnem pomnilniku. Njegova kapaciteta je odvisna od strojne opreme oz. grafične kartice. V začetnem obdobju proizvodnje igralnih aparatov v podjetju leta 2006 je tak pristop zadostoval potrebam trga in zahtevam animatorjev. Strojna oprema je bila za takratne potrebe dovolj zmogljiva, danes pa žal ni več tako zaradi vse večjega števila, dolžine in kompleksnosti animacij. Poleg težav s pomanjkanjem grafičnega pomnilnika so se tudi povečali časi nalaganja vseh potrebnih slik v grafični pomnilnik, ki znašajo tudi do 30 sekund in več. Predolg čas nalaganja negativno vpliva na igralčevo izkušnjo, saj se igralec naveliča čakanja in odide h konkurenčnejšim igram. Odpravljanja teh slabosti se bomo lotili na primeru grafično najzahtevnejše igre slot, za katero je izdelanih že preko 30 grafičnih tem.

4.2 Izbira rešitve

Težave s pomanjkanjem grafičnega pomnilnika so rešljive z zmogljivejšo strojno opremo, vendar grafični oblikovalci že sedaj presegajo njene zmogljivosti, kar zahteva drugačne rešitve v grafičnem podsistemu igre. Možnosti so:

- pretakanje slik (ang. texture streaming),
- predvajanje video posnetka,
- uporaba 3D animacije,
- izboljšava algoritma.

Pretakanje slik pomeni sprotno nalaganje slike za sliko v grafični pomnilnik med prikazovanjem animacije, prejšnjo že prikazano sliko pa nato takoj odstranimo iz pomnilnika. S tem drastično zmanjšamo čas začetnega nalaganja in porabo grafičnega pomnilnika. Rešitev je enostavna in hitro izvedljiva, vendar se je v testni izvedbi izkazala za neuporabno zaradi prepočasnega nalaganja slik. Razlog

počasnosti je uporabljen zapis (format) slik in velikost datotek, ki sta neprimerna za tak način uporabe. Slike so zapisane v formatu PNG (Portable Network Graphics, 2012), ki je zelo razširjen in ga uporabljamo zaradi zmožnosti hranjenja informacije o transparentnosti (ang: alpha channel). Za primerjavo: slike podobno razširjenega formata JPEG (JPEG, 2012) so sicer manjše po velikosti, vendar žal ne morejo biti transparentne in ga zato ne uporabljamo. Uporabljene funkcije v Microsoftovi platformi .NET (DePetrillo, 2002) za nalaganje in dekodiranje PNG slik so prepočasne, zato se operacije ne izvajajo v realnem času, posledica pa je zatikajoča se animacija. S takšnim pretakanjem slik bi uporabnik videl le nekaj (3 do 5) slik na sekundo, kar je daleč pod zahtevano mejo hitrosti 25 slik na sekundo (Frame rate, 2012).

Postopek predvajanja video posnetka je podoben kot pretakanje slik, vendar ob uporabi video kodeka (Video codec, 2012). To je zahteven namenski algoritem, ki video posnetek v stisnjeni obliki razširi oz. dekodira in generira vsebovane slike. Največkrat je operacija tudi strojno pospešena s pomočjo grafične kartice, kar zelo poveča hitrost. Sposobnost predvajanja video posnetkov je že na voljo v dosedanjem grafičnem pogonu aparata slot in jo tudi uporabljamo za celozaslonske animacije med pomembnejšimi fazami igre. Predvajanje je zahtevnejše opravilo (AVI Video textures, 2012), vendar enako kot velja za JPEG slike, tudi video posnetki oziroma velika večina njihovih kodekov (Popular Video Codecs, 2012) žal ne podpira transparentnosti (alfa kanal). Brez tega pa ne moremo prikazati npr. barvnih balonov, ki letijo nad dinamično vsebino igre.

Implementacija in uporaba 3D animacij v grafičnem pogonu aparata slot sta zahtevni, vendar na osnovi izkušenj in poznavanja programske strukture igre ocenjujemo, da bi izvedba trajala predolgo. Dodatno se poveča zapletenost izvedbe zaradi izdelave animacij in specifične uporabe objektov v igri. Vprašljiva je tudi smiselnost uporabe 3D tehnike za dosedanje relativno enostavne animacije, končno pa je le-ta nesmotrna zaradi drugačnih načrtov razvojnega oddelka v podjetju.

Kot edina smiselna rešitev tako ostaja le še izboljšava dosedanjega algoritma računalniške animacije. Ta bi torej morala zagotoviti večjo učinkovitost animacij tako, da bi:

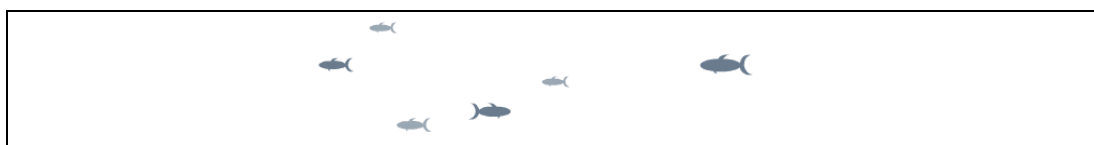
- zmanjšala porabo grafičnega pomnilnika,
- skrajšala časa nalaganja slik v grafični pomnilnik.

Rešitev je v zmanjšanju števila slik in skupne slikovne površine do te mere, da bo čas nalaganja teme igre slot občutno krajši, npr. manj kot 15 sekund, in zato sprejemljiv za igralce. Takšna pohitritev je naš cilj, ki ga bomo skušali doseči z izboljšavo algoritma.

5 IZBOLJŠAVA ALGORITMA

5.1 Zasnova

Med pregledovanjem animacij v igri slot smo ugotovili, da vsebujejo veliko slik z večjimi prazninami oz. s transparentnimi točkami, ki zasedajo omejen grafični pomnilnik. Tipičen primer vidimo na sliki 18: na celotni sliki je le nekaj objektov (rib), ostale točke v sliki pa so nepotrebne, saj ne predstavljajo nobene vsebine oz. objektov.



Slika 18: Slika animacije z veliko praznine okrog objektov

Če bi otroku dali nalogo, naj nariše ribe na prozorno folijo, bi jih narisal, ne bi pa barval preostale površine, ker je to nepotrebno in nesmiselno. Enako nesmiselno je v sliki animacije hraniti in nato izrisovati transparentne točke, saj ne spremenijo končne slike na zaslonu, ki jo vidi uporabnik. Smiselno je hraniti sliko, ki vsebuje le potrebne objekte, in jih na želeno mesto postavljali med predvajanjem animacije. To idejo ilustrirajmo s primerom, ki ilustrira možen prihranek pomnilniškega prostora: izvorna slika 18 je sestavljena iz 131.072 točk (1024×128), slika 19, ki vsebuje le objekte, potrebne za animacijo, pa iz 4.048 točk (92×44), kar je le 3 % izvornika.



Slika 19: Ideja rešitve: slika animacije le s potrebnimi objekti

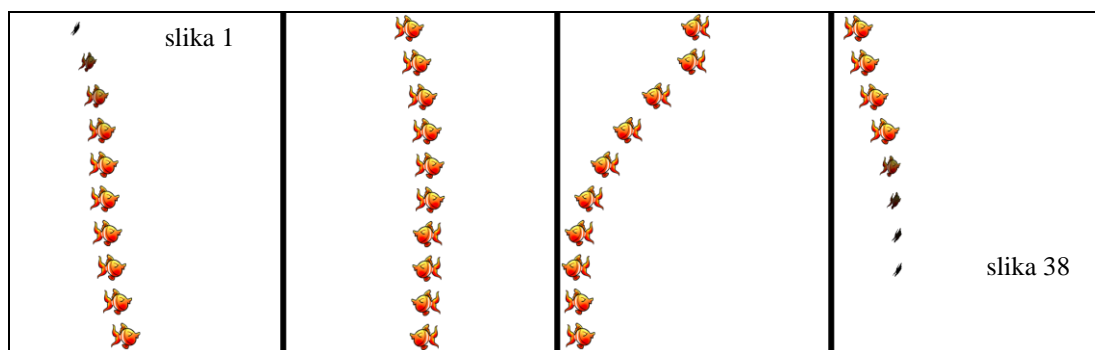
To je zelo spodbuden rezultat. Objekti sami niso dovolj, potrebujemo še podatke o njihovih položajih na zaslonu. Iz izkušenj vemo, da taki podatki zasedajo zelo malo dodatnega pomnilniškega prostora v primerjavi z vsemi slikami animacije.

V prvi fazi potrebujemo torej postopek detekcije in izrezovanja koristnih vsebin (objektov) iz slik animacije, s katerim bomo odstranili praznine ter prihranili prostor

v grafičnem pomnilniku. V drugi fazi pa bomo s postavljanjem izrezanih objektov v eno sliko še dodatno skrajšali čas nalaganja slik v grafični pomnilnik.

5.2 Izrezovanje koristne vsebine animacije

Za primer vzemimo animacijo premikajočega se objekta (plavajoče ribe) na sliki 20.



Slika 20: Vse slike animacije premikajočega se objekta

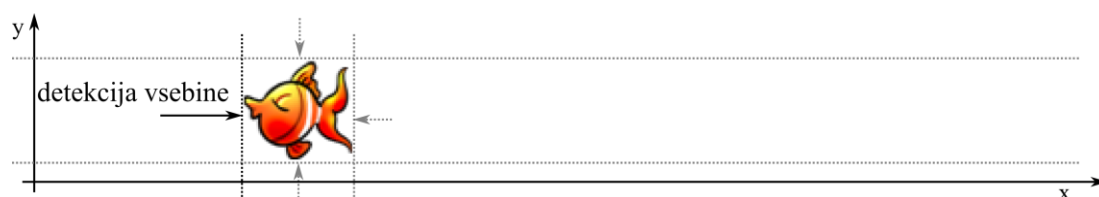
Izbrana animacija je dovolj enostavna za razlago algoritma: v vsaki sliki je le po en objekt (slika 21), ki ga bomo izrezali.



Slika 21: Primer slike animacije premikajočega se objekta

Ker pri obdelavi slik najlažje razumemo in implementiramo postopke, ki manipulirajo s stolpci in vrsticami točk ter pravokotnimi oblikami s stranicami po oseh x in y , tudi uporabimo te elemente pri zasnovi algoritma.

Detekcijo vsebine izvajamo po oseh x in y iz vseh strani proti sredini slike, kot je to prikazano na sliki 22.



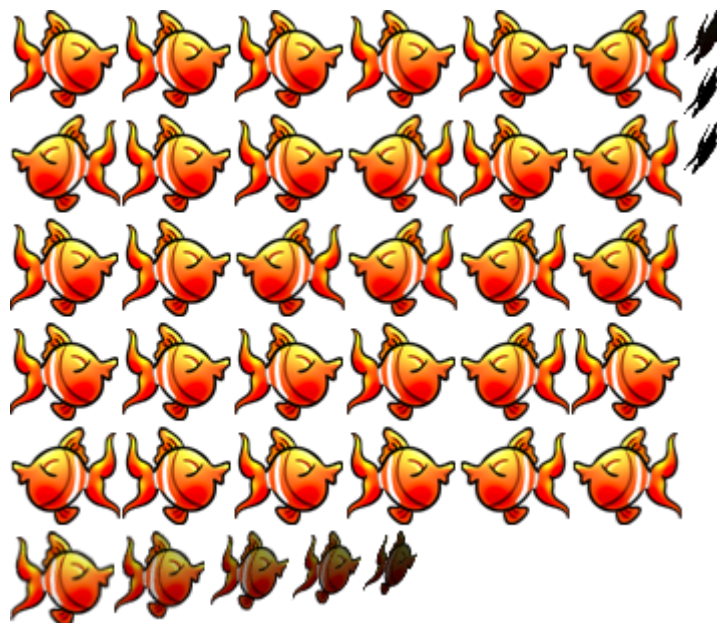
Slika 22: Detekcija vsebine po oseh x in y

Detekcija poteka tako, da najprej preverjamo vse točke vzporedne osi y od leve proti desni. Ko zaznamo neprazen stolpec oziroma točko, ki ni popolnoma transparentna, se ustavimo, v nasprotnem primeru nadaljujemo preverjanje naslednjega stolpca proti desnemu robu slike. Detekcijo nato ponovimo še iz nasprotne smeri, od desne proti levi, ter enako od zgoraj in spodaj. Rezultat je izrezan objekt, prikazan na sliki 23.



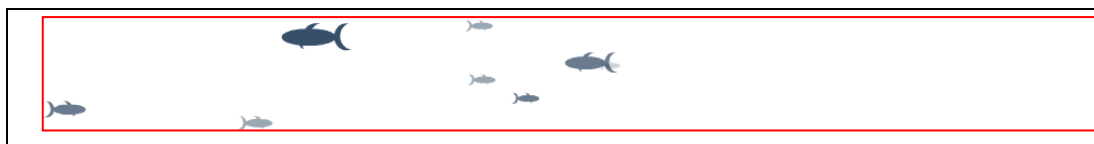
Slika 23: Izrezan objekt

Če izrezovanje izvedemo na vseh 38 slikah animacije, prikazanih na sliki 20, pridobimo samostojne objekte, ki so prikazani na sliki 24.



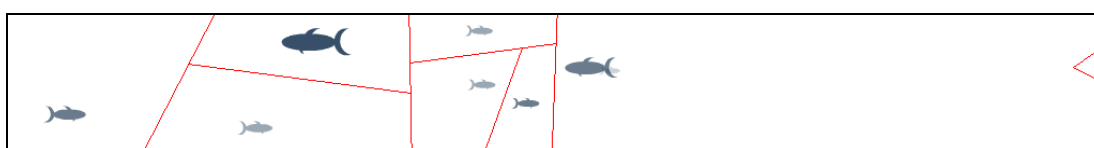
Slika 24: Izrezani objekti celotne animacije

Do sedaj smo obravnavali enostavne slike oz. slike s po enim objektom. Ker smo med pregledovanjem animacij ugotovili, da animacije v večini primerov vsebujejo več objektov na sliki, predvidevamo, da enostavno izrezovanje ne bo dovolj. Kot primer vzemimo rezultat enostavnega izreza, prikazanega na sliki 25.



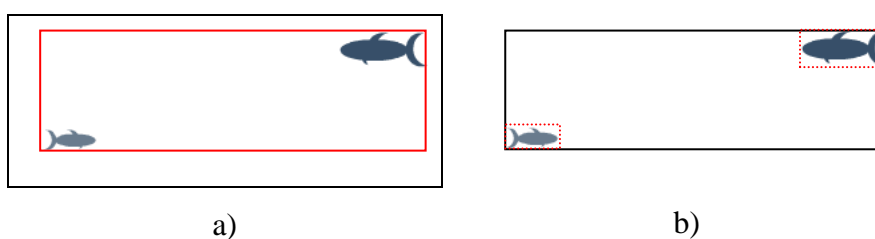
Slika 25: Enostavno izrezovanje v primeru zelo razpršenih objektov po sliki

Na sliki 25 je vidna očitna slabost takega izrezovanja. Potrebujemo torej način izrezovanja objektov znotraj objekta, dobljenega z enostavnim izrezovanjem. Iz papirja bi verjetno s škarpami začeli izrezovati nekako tako, kot je prikazano na sliki 26.



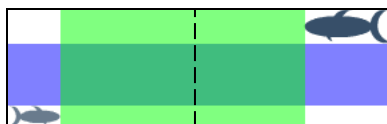
Slika 26: Primer izreza objektov s škarpami

Podoben pristop lahko uporabimo tudi pri zasnovi postopka izrezovanja. Za razlago vzemimo primer dveh problematičnih objektov, ki ležita v nasprotnih kotih (slika 27a). Rezultat izrezovanja (slika 27b) je objekt, sestavljen iz dveh objektov, med njima pa je še vedno veliko praznine.



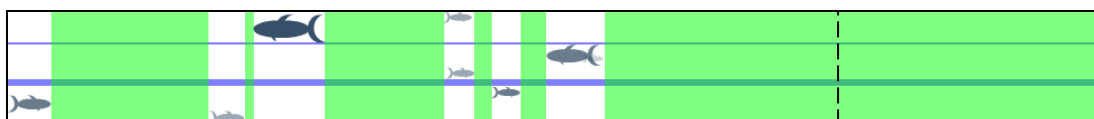
Slika 27: Enostavno izrezovanje slike z objektoma v nasprotnih kotih: a) izvorna slika, b) rezultat enostavnega izreza

Izrezan objekt je zato smiselno v naslednjem koraku razdeliti na dva dela in ponoviti postopek izrezovanja. Ker manipuliramo le s pravokotnimi oblikami, lahko razdelimo sliko samo vodoravno ali navpično – vendar kje? V naslednjem koraku razdeljevanja poiščemo vodoravne (modre) in navpične (zelene) pasove praznin (slika 28) ter izberemo tistega, ki ima največjo površino. Sliko razdelimo in posledično s tem odstranimo največ praznine v tem koraku.



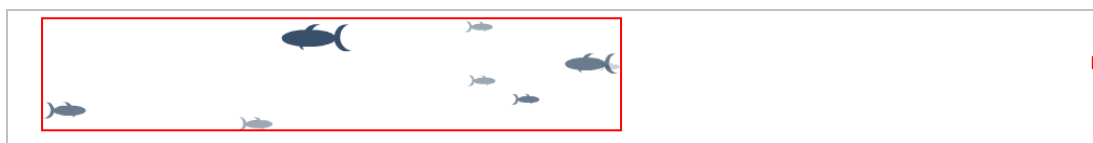
Slika 28: Pasovi praznin in izbrani rez

Enak princip velja, ko je objektov na sliki več, vendar je v tem primeru možnih pasov več. Primer izbranega reza za sliko z več objekti je prikazan na sliki 29.



Slika 29: Več objektov, več praznih pasov

Po izbiri reza sliko razdelimo na dva dela po osi x ali y, nad slikama pa ponovno izvedemo korak enostavnega izreza. Rezultat tega postopka je prikazan na sliki 30.



Slika 30: Rezultat delitve slike in ponovnega izreza

Koraka izreza in delitve slike lahko ponavljamo, dokler ni več praznih pasov ali pa dokler število korakov ne preseže določene meje. Rezultat takega izrezovanja objektov iz slike 25 vsebuje le 4,6 % točk (slika 31).



Slika 31: Izrezani objekti

5.3 Odstranjevanje enakih objektov

Po izrezovanju različnih testnih animacij ugotovimo, da so si izrezani objekti mnogokrat med seboj zelo podobni. Že med bežnim ogledom slike 24 opazimo veliko podvojenih objektov. V naslednjem koraku je zato smiselno odstraniti dvojnike s primerjanjem vseh objektov med seboj. Končni rezultat te operacije je

slika 32 resolucije 140×136 oz. skupno 19.040 točk, kar je le 1,5 % vseh slikovnih točk izvirne animacije prikazane na sliki 20.



Slika 32: Končni rezultat izrezovanja objektov z odstranjenimi dvojniki

5.4 Postavitev izrezanih objektov v eno sliko

V prvi fazi izboljšane algoritma animacije smo izrezali koristne vsebine, ki se uporabljajo v animaciji. S tem smo odstranili nepotrebne praznine v slikah in pridobili več manjših slik oz. objektov. Tako smo zmanjšali skupno število slikovnih točk, potrebnih za hranjenje vsebine animacije, kar je eden izmed ciljev, nismo pa zmanjšali oz. smo celo povečali skupno število slik, ki jih je potrebno naložiti v grafični pomnilnik. V naslednjem koraku moramo zato zmanjšati število slik s pakiranjem objektov v eno ali več slik.

Postavljanje pravokotnih elementov na čim manjšo površino je poznan problem. Že z iskanjem po spletu s ključnimi besedami, kot so »bin packing problem« (Bin-Packing Problem, 2012) ali »texture packing«, dobimo veliko postopkov reševanja (Lodi in drugi, 2002), od enostavnih z razvrščanjem (Packing lightmaps, 2012) do zahtevnejših. Ta problematika je še posebej zanimiva in se želimo soočiti z njo, zato smo tudi lotili razvoja postopka za njeno reševanje.

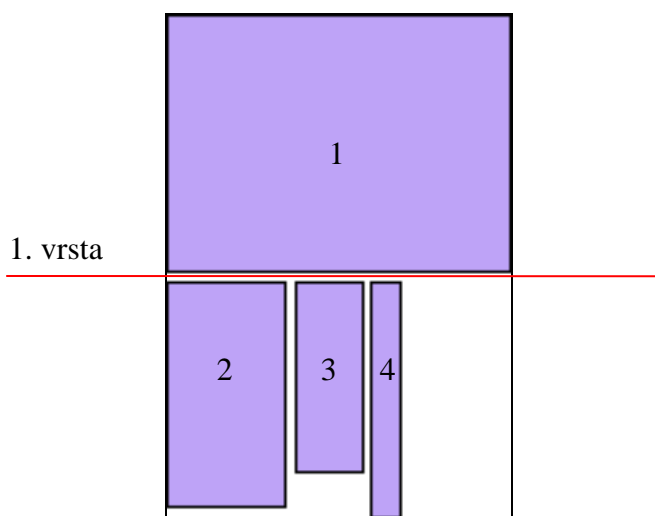
Problem vsebuje dve medsebojno odvisni vprašanji:

- na kakšen način polagati pravokotnike (objekte) – različne postavitve objektov namreč vplivajo na končno velikost površine (slike),
- kako veliko površino uporabiti – velikost površine vpliva na postavitev objektov.

Odločili smo se razviti postopek postavljanja objektov na površino dane velikosti ter postavitev nato iterativno spreminjati v upanju, da najdemo še manjšo potrebno površino. Med razvojem smo uporabili testne objekte, izrezane iz animacij, ki so se razlikovale po številu slik in objektov v njih, po tem, da so vsebovale objekte podobnih in različnih velikosti ter objekte različnih razmerij (pokončni in vodoravni pravokotniki). S takim naborom objektov smo preizkušali postopek postavljanja objektov na površino v različnih pogojih.

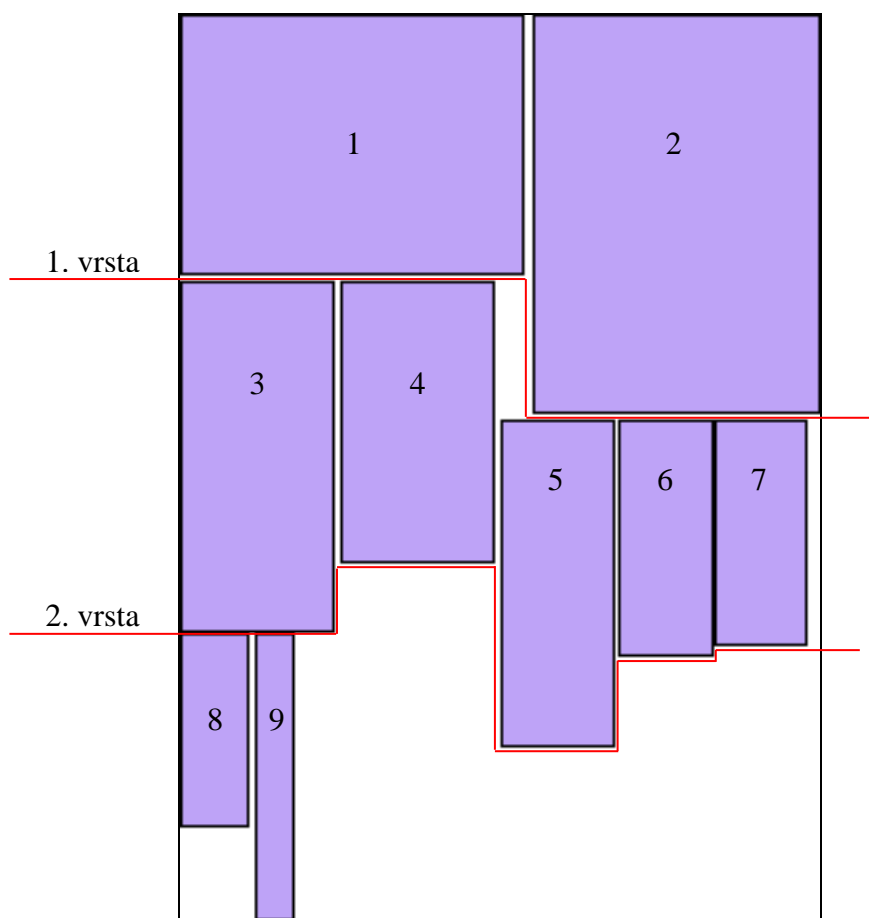
Prva in najhitrejša implementacija rešitve je enostavno postavljanje objektov v vrste kot besede v knjigi v vseh možnih permutacijah. Žal se število permutacij s številom objektov zelo povečuje, tako da pristop v praksi za več kot 12 objektov ni smisel, saj se čas rešitve poveča že na nekaj 10 minut. Za 12 objektov je na voljo $12! = 479.001.600$ permutacij oz. različnih postavit, za 14 elementov pa že $14! = 87.178.291.200$. Število objektov v animacijah elektronskih iger, kakršne obravnavamo, pa je ponavadi več kot 10, zato je taka rešitev neuporabna.

Za bolj smiselno se izkaže postavljanje objektov po vzorcu zlaganja knjig na kup: velike spodaj, majhne zgoraj. To načelo smo zato uporabili za postopek postavljanja, vendar postavljanje začnemo zgoraj z najširšim objektom, nadaljujemo desno kot pisanje besedila in končamo z najožjim objektom. Primer vidimo na sliki 33, kjer številke objektov označujejo vrstni red postavljanja. Za enostavne primere z nekaj objekti je tako postavljanje zadovoljivo, težave pa nastopijo pri več objektih.



Slika 33: Osnovna zamisel postavljanja objektov v sliko

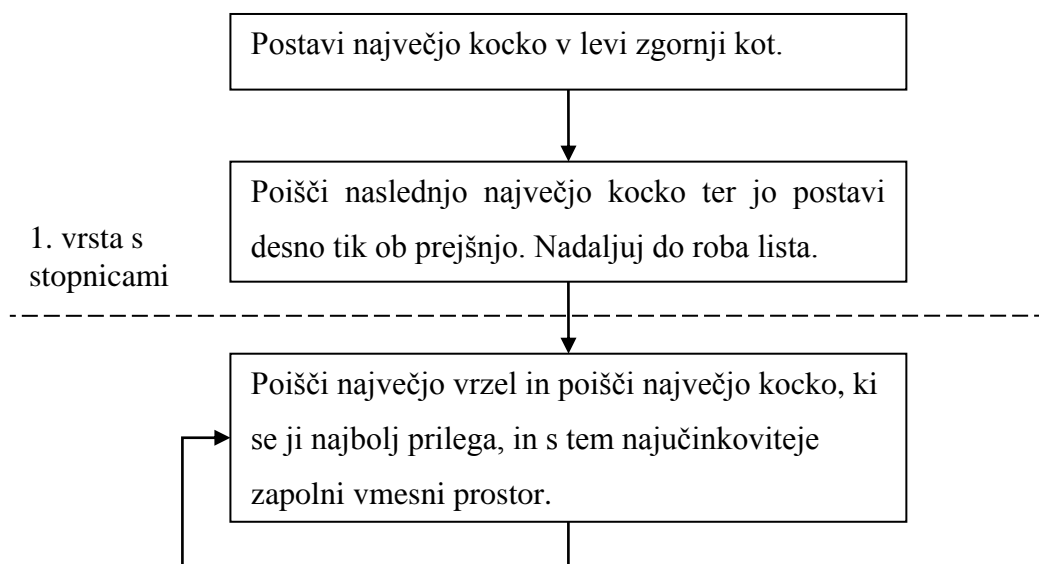
Enostavno zaporedno postavljanje objektov od širših proti ožjim po vrstah od leve proti desni (kot besedilo v knjigi) ne prinaša zadovoljivih rezultatov zaradi stopničavosti, ki se pojavlja v postavitvah. To težavo ilustriramo s primerom na sliki 34. Ob njenem ogledu bi verjetno predlagali premik objektov 5 in 7 na mesto objekta 4, vsekakor pa objektov 8 in 9 pod objekt 4. Postavljanje večjega števila objektov v več vrst brez takšnih popravkov povzroča vse večjo stopničavost, kar postane neuporabno. Ugotavljamo torej, da tako enostaven postopek postavljanja ni ustrezen, saj ne prepozna vrzeli, kakršna v našem primeru nastane pod objektom 4.



Slika 34: Stopničavost pri postavitvi več objektov v sliko

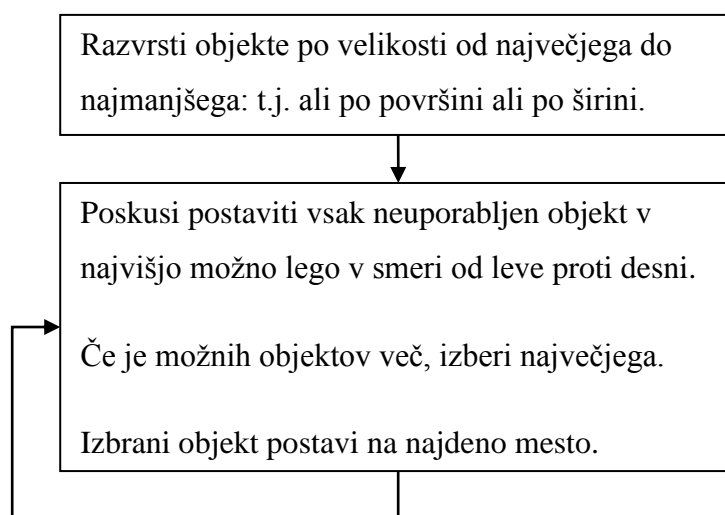
Postopek postavljanja očitno potrebuje nadaljnje izboljšave, vendar ne želimo dodajati posebnih postopkov za neugodne primere, kot je omenjena stopničavost. Programska koda bi v tem primeru postala preobsežna in zato prezahtevna za vzdrževanje. Glede na poznavanje podobnih problematik in izkušenj sklepamo, da je smiselna rešitev kombinacija dveh skrajnih pristopov: zaporednega postavljanja v vrsto in poskušanja vseh možnih kombinacij.

Do enostavne in razumljive rešitve lahko pridemo z razmislekom. Predstavljamo si, da moramo rešiti nalogo zlaganja različno velikih kock po površini lista papirja velikosti A4. Miselni proces bi najbrž bil tak, kot je prikazan na sliki 35.



Slika 35: Miselni proces med postavljanjem kock

Na podlagi miselnega procesa sestavimo postopek postavljanja pravokotnih objektov na površino, prikazan na sliki 36.



Slika 36: Postopek postavljanja pravokotnih objektov na površino

Primer rezultata postavljanja po tem postopku vidimo na sliki 37.



Slika 37: Rezultat postavljanja pravokotnih objektov na površino po različnih kriterijih razvrščanja: a) po površini, b) po širini

Na sliki 37a je objekt 9 na neugodnem mestu predvsem zaradi objekta 8. Manjšo zasedenost površine dosežemo s premikom objekta 8 pod objekt 4. Z razvrščanjem po širini pa s postopkom postavljanja uspešno postavimo objekt 4 v vrzel med objekti 1, 2 in 3 (slika 37b). Postavitev na sliki 37a zasede 5 % več površine kot postavitev na sliki 37b.

Po testiranju postopka postavljanja s slikami različnih animacij smo ugotovili, da nobeden od dveh kriterijev razvrščanja ni bil v splošnem najboljši. V nekaterih primerih so boljši rezultati z razvrščanjem objektov po površini, v drugih primerih pa po širini.

6 OVREDNOTENJE IZBOLJŠANEGA ALGORITMA

6.1 Poraba grafičnega pomnilnika

Porabo grafičnega pomnilnika smo želeli zmanjšati z izrezovanjem oz. odstranjevanjem praznih delov slik animacije. Med sprotnim vrednotenjem v času razvoja algoritma je izrezovanje dalo zelo dobre rezultate: rešitev na sliki 19 obsega le 3 % točk glede na izvorno sliko. Dober rezultat smo dosegli tudi v primeru animacije premikajočega se objekta (ribe) s slike 20, saj dobljena slika izrezanih objektov obsega le 8 % točk izvornih slik (tabela 1).

Tabela 1: Prostorske izboljšave po izrezovanju slik premikajočega se objekta

Animacija plavajoče ribe	Resolucija slik	Število slik	Skupno število točk
Izvirne slike animacije	512×64	38	1.245.184
Slika izrezanih objektov	356×308	1	109.648

Postopek izrezovanja objektov smo izvedli na več kot 30 temah igre slot. Nekaj podatkov je zbranih v tabeli 2, v kateri vidimo zasedenost medpomnilnika z grafičnimi temami, ki so pripravljene za prenos v grafični pomnilnik, pred izrezovanjem slik in prihranek predpomnilnika po izrezovanju. V predpomnilniku se sicer nahajajo tudi druge slike, ki niso del animacij, kot npr. slike ozadja, vendar je teh v primerjavi s slikami animacij malo.

Prostorski prihranek medpomnilnika po izrezovanju na posameznih preizkušanih temah igre slot se giblje med 42 % in 83 %, vse teme skupaj pa zasedajo slabih 40 % začetne velikosti, kar pomeni prihranek dobrih 60 %. Iz tega lahko sklepamo, da so slike animacij v dosedanji obliki vsebovale več kot polovico praznega prostora.

Tabela 2: Zasedenost predpomnilnika z grafičnimi temami pred izrezovanjem slik in prihranek po njem

Tema igre	Zasedenost predpomnilnika pred izrezovanjem [MB]	Prihranek predpomnilnika po izrezovanju [%]
FrogPrinceW	347	76
JuicyFruitsW	112	54
FarmManiaW	237	72
LotusPrincessW	117	48
GoldenReefW	424	78

6.2 Izračunani čas nalaganja slik

Čas nalaganja slik v grafični pomnilnik je odvisen od števila in velikosti slik. V uporabi je klasičen proces nalaganja slik, ki v igri slot obsega naslednje zaporedje korakov:

1. grafični pogon dostopi do pomnilniškega medija (trdi disk),
2. prebere sliko,
3. pretvori sliko v interni format,
4. prenese v grafični pomnilnik.

Čas dostopa do podatkov, v tem primeru slike, na trdem disku, je odvisen od mehanskih lastnosti diska, kot sta hitrosti vrtenja plošč in hitrosti premikanja bralne glave. Po uspešni postavitvi bralne glave se začne branje podatkov. Hitrost branja pa ni odvisna le od lastnosti diska, ampak tudi od razdrobljenosti datoteke (File system fragmentation, 2012), ker mora bralna glava s premiki dostopati do vseh delov zapisa

datoteke na disku, kar terja dodaten čas. V izračunih smo predpostavljali, da datoteke niso fragmentirane.

Dostopanje do slik in njihovo branje pa ni odvisno le od mehanskih lastnosti trdega diska in razdrobljenosti datotek, ampak tudi od internih stanj operacijskega sistema, predvsem od predpomnilnika datotečnega podsistema. V njem se namreč lahko že nahajajo želene slike, tako da dostop do diska ni potreben in aplikacija takoj pridobi podatke. Ker igra slot zasede večino glavnega pomnilnika in med delovanjem dostopa tudi do drugih datotek aplikacije, ki jih sistem ravno tako zadrži v datotečnem predpomnilniku, prejšnje datoteke, npr. slike predhodne teme, zaradi omejenosti predpomnilnika zavrže. Zaradi takega delovanja datotečni predpomnilnik največkrat žal ne skrajša časa menjave teme oz. nalaganja slik, saj želenih slik ne vsebuje več.

Nalaganje slik se tipično izvaja med menjavo teme, ki jo lahko sproži igralec po zaključku trenutne igre. Za enostavnejši prikaz časovnih okvirov smo izračunali povprečen čas dostopa do vseh slik (tabela 3) in čas njihovega branja (tabela 4). Ker je v igri slot v uporabi trdi disk nižjega cenovnega razreda in zmogljivosti v rangi modela Western Digital WD2500JD (SATA Hard Drives Run Riot, 2012), smo kot povprečen dostopni čas v izračunih privzeli 15 ms (HDD Charts 2012: Read Access Time, 2012), kot povprečno hitrost branja pa 75 MB/s (HDD Charts 2012: Read Throughput Average, 2012).

Izračuni kažejo, da se zaradi zmanjšanja števila slik skupni dostopni čas občutno skrajša, z nekaj 10 sekund na le 2 sekundi. Število slik na temo igre smo zmanjšali s prvotnih 400 do 1.400 na 77 do 220 slik, skupni dostopni čas na temo pa s prvotnih 6 do 21 sekund na 1,6 do 3,3 sekunde.

Tabela 3: Izračunani dostopni čas do slik pred izboljšavo in po njej

Tema igre	Pred izboljšavo		Po izboljšavi	
	Število slik	Skupni dostopni čas [s]	Število slik	Skupni dostopni čas [s]
FrogPrinceW	1.420	2,1	178	2,7
JuicyFruitsW	525	7,9	97	1,5
FarmManiaW	1.326	19,9	146	2,2
LotusPrincessW	761	11,4	135	2,0
GoldenReefW	2.595	38,9	150	2,3

Tabela 4: Izračunani čas branja vseh slik pred izboljšavo in po njej

Tema igre	Pred izboljšavo		Po izboljšavi	
	Prostorska zasedenost [MB]	Čas branja slik [s]	Prostorska zasedenost [MB]	Čas branja slik [s]
FrogPrinceW	347	4,6	82	1,1
JuicyFruitsW	112	1,5	51	0,7
FarmManiaW	237	3,2	66	0,9
LotusPrincessW	117	1,6	61	0,8
GoldenReefW	424	5,7	94	1,3

Zmanjšanje prostorske zasedenosti pripomore k hitrejšemu delovanju, vendar ne v taki meri kot zmanjšanje števila slik. Kljub temu pa smo skupni čas nalaganja slik uspeli bistveno skrajšati (tabela 5).

Tabela 5: Izračunani skupni čas nalaganja slik pred izboljšavo in po njej

Tema igre	Pred izboljšavo [s]	Po izboljšanju [s]
FrogPrinceW	6,8	3,8
JuicyFruitsW	9,4	2,2
FarmManiaW	23	3,1
LotusPrincessW	13	2,8
GoldenReefW	45	3,6

Po branju in prenosu slike v sistemski pomnilnik je potrebno sliko najprej pretvoriti v obliko, ki je primerna za strojno opremo oz. grafično kartico (korak 3 v procesu nalaganja slik), preden se jo prenese v grafični pomnilnik. Grafične kartice znajo prikazovati le nekaj oblik slik: surove, kot je npr. RGB, ali stisnjene S3TC (S3 Texture Compression, 2012) in njim podobne. Potrebna pretvorba je zamudno opravilo in lahko traja tudi minuto ali dve za vsako temo igre, izvede pa se le po potrebi oz. samo enkrat po zagonu igralnega aparata. Pretvorjene slike se hranijo v medpomnilniku aparata (na trdem disku), zato tega časa nismo upoštevali, saj se pretvorba ne izvaja med rednim potekom igre.

Zadnji korak nalaganja slik je prenos slik v pretvorjeni obliki v grafični pomnilnik. Prenos med sistemskim pomnilnikom in grafičnim pomnilnikom je zelo hiter, saj je to pogosto in pomembno opravilo v svetu grafičnih pogonov in strojne opreme, giblje pa se v območju nekaj GB/s (GPU Memory Transfer, 2012). Za prenos take količine podatkov kot jo zahtevajo slike najboljše teme GoldenReefW pred izboljšavo, ki zaseda 424 MB, potrebuje računalnik le približno 200 ms. V praksi je

ta čas večji, saj je potrebno prenesti okoli 2.600 slik, oz. celotno operacijo prenosa ponoviti 2.600-krat, vsekakor pa prenos naj ne bi trajal več kot nekaj sekund. V primerjavi s skupnimi 45 sekundami nalaganja prej omenjene teme je ta čas minimalen.

Med drugim je splošno pravilo pisanja programske opreme za grafično opremo to, da manj kot je komunikacije z njo, hitreje sistem deluje. Z zmanjšanjem števila slik smo tudi z upoštevanjem tega pravila pohitrili prenos slik v grafični pomnilnik.

6.3 Izmerjeni čas nalaganja teme na igralnem aparatu

Želeli smo tudi preveriti, če izboljšani algoritem računalniške animacije v igralnem aparatu v resnici pohitri nalaganje, kot kažejo izračuni. Meritve smo izvajali na enem izmed aparatov v testnem laboratoriju podjetja.

Igra oz. njene grafične teme imajo sicer nekatere posebnosti glede uporabe slik in animacij, ki niso natančna preslikava predpostavk, uporabljenih v izračunih v prejšnjem podpoglavju, a bi njihova razlaga presegla obseg in namen diplomskega dela. Ne glede na posebnosti ostaja cilj enak: skrajšati čas nalaganja tem na sprejemljivo mejo za uporabnika.

Največkrat je tema razdeljena na dva dela: glavni in bonus del. Glavni del se nalaga vedno ob zamenjavi teme, bonus pa večkrat med igro izbrane teme. Posebej smo merili čas nalaganja glavnega dela in posebej bonus dela. Ker je igralni aparat dinamičen sistem in se na njem med nalaganjem slik izvajajo tudi drugi procesi znotraj igre in ostalih aplikacij, pričakujemo odstopanja od izračunanih vrednosti, vsekakor pa pohitritve. Meritve smo skušali izvajati v čimbolj podobnih pogojih kot so bile predpostavke v izračunih, npr. s praznim predpomnilnikom datotečnega podsistema. Navkljub vsemu so meritve na igralnem aparatu pokazale zelo dobre rezultate, ki so zbrani v tabeli 6.

Tabela 6: Izmerjeni časi nalaganja tem v igri slot pred izboljšavo in po njej

Tema igre	Pred izboljšavo [s]		Po izboljšavi [s]	
	Glavni del	Bonus del	Glavni del	Bonus del
FrogPrinceW	17,8	20,0	6,3	2,0
JuicyFruitsW	17,7	–	6,0	–
FarmManiaW	19,7	15,0	12,7	2,7
LotusPrincessW	12,7	10,0	8,3	4,0
GoldenReefW	41,7	33,3	15,7	4,7

Rezultati so presegli naša pričakovanja kot tudi pričakovanja sodelavcev v oddelku. Čas nalaganja smo občutno skrajšali in dosegli sprejemljive vrednosti, v nekaterih primerih je čas krajši celo za 10-krat. Z izboljšanim algoritmom računalniške animacije tako dosegamo želene rezultate in ga smatramo kot ustrezno rešitev obravnavane problematike.

7 ZAKLJUČEK

V diplomskem delu smo obravnavali problematiko, ki je nastala zaradi uporabe vse večjega števila in naraščajočih dolžin animacij v temah igre slot, to je primanjkovanja grafičnega pomnilnika in dolgega časa nalaganja slik. S poznavanjem vrst računalniških animacij, analizo delovanja igre in uporabljenih animacij ter izbiro primerne rešitve smo razvili izboljššan algoritem grafične animacije za igro slot.

Uvodoma smo spoznali sestavo digitalne rastrske slike kot temelj računalniške grafike ter nekaj značilnih vrst animacij. Opisali smo film kot osnovno vrsto, nadaljevali s sestavljeno večslojno animacijo, zaključili pa z zahtevno 3D animacijo.

K problematiki smo pristopili z analizo dosedanjega algoritma računalniške animacije v igri in tistih njegovih značilnosti, ki so bistven vzrok težav. Glede na poznavanje programske opreme smo imeli na voljo več možnih rešitev: predvajanje animacij s pretakanjem slik, uporabo video posnetkov, uporabo 3D animacij in razvoj izboljššanega algoritma. Izbrana rešitev je bila slednja, ostale namreč niso bile primerne zaradi prezahtevnosti, neustreznosti ali drugih zahtev v podjetju.

Po analizi animacij in ugotovitvi, da slike animacij vsebujejo veliko praznega prostora, ki po nepotrebnem zaseda omejen grafični pomnilnik, smo idejno zasnovali izboljšavo algoritma. Cilj je bil zmanjšati zasedenost grafičnega pomnilnika in pohitrili nalaganje slik za animacije vanj.

Manjšo porabo pomnilnika smo želeli doseči s postopkom izrezovanja koristne vsebine animacije. Najprej smo določili korak izrezovanja na primeru enostavne slike s po enim objektom, nadaljevali pa z izrezovanjem slike z več objekti in ugotovili, da ta korak ne zadošča. Postopek smo razširili s korakom razdeljevanja slike na dva dela in s ponavljanjem obeh korakov uspešno izrezali vse koristne objekte v slikah animacije. Ugotovili smo, da je veliko objektov med seboj enakih in postopek zaključili s smiselnim korakom izločanja podvojenih objektov. S takšnim postopkom izrezovanja koristne vsebine smo uspešno prihranili veliko grafičnega pomnilnika.

Z izrezovanjem koristne vsebine smo pridobili veliko manjših slik, bistveno več kot je izvornih slik animacije. To bi znatno povečalo čas nalaganja grafične teme igre, česar pa ne želimo. To težavo smo rešili z razvojem postopka postavljanja izrezanih objektov (slik) v skupno sliko. Ta problem postavljanja vsebuje dva med seboj odvisna izziva: način postavljanja vpliva na potrebno velikost skupne slike in obratno. V prvem koraku razvoja smo postavljali objekte v vrste kot besedilo v knjigi s preverjanjem možnih permutacij, vendar se je to izkazalo za neuporabno, saj je postavljanje več kot 12 objektov časovno prezahtevno. Drugi korak je bil podoben prvemu, le da smo objekte postavljali po zaporedju od najširšega do najožjega. Dobljene postavitve niso bile zadovoljive zaradi pojavljanja stopničavosti in vrzeli med objekti, zato algoritem ni ustrezno postavil objektov oz. zapolnil praznin v končni sliki. V tretjem koraku smo združili oba osnovna pristopa, tj. preverjanje vseh permutacij in postavljanje objektov kot besedila v knjigi. Na ta način smo se uspešno izognili stopničavosti, zadovoljivo postavili izrezane objekte v sliko in s tem zmanjšali število slik animacije na samo eno sliko.

Izboljšani algoritem računalniške animacije, sestavljen iz postopka izrezovanja koristne vsebine in postavljanja izrezanih objektov v eno sliko, smo po njegovi programski izvedbi ovrednotili in preverili, ali ustreza zadanim ciljem: zmanjšanju porabe grafičnega pomnilnika in skrajšanju časa nalaganja slik vanj. Najprej smo ovrednotili zasedenost pomnilnika po izrezovanju. Že med razvojem postopka izrezovanja smo dosegali dobre rezultate, na igralnem aparatu pa smo v povprečju prihranili okrog 60 % pomnilnika, potrebnega za hranjenje slik grafične teme igre, v najboljšem primeru pa tudi več kot 80 %. Nato smo vrednotili še čas nalaganja slik na dva načina, prvič z izračuni in upoštevanjem bistvenih vplivnih dejavnikov, kot so lastnostni strojne opreme, operacijskega sistema in načina delovanja igre, drugič pa z merjenjem časa nalaganja na igralnem aparatu. Kljub temu, da v aparatu poleg igre potekajo tudi druga opravila, so se glavni deli tem nalagali v povprečju dvakrat hitreje, bonus deli šestkrat, nekateri pa tudi do desetkrat hitreje.

S prihrankom grafičnega pomnilnika in pohitritvijo nalaganja tem lahko neposredno vplivamo na grafično pestrost in privlačnost elektronske igre. Če vzamemo primer teme GoldenReefW v igri slot, kjer se v ozadju prikazujejo animacije z morskega dna, ga lahko z izboljšanim algoritmom animacije močno popestrimo. Z dodajanjem

novih animacij postane morsko dno živahnejše in bolj raznoliko, brez da bi se ob tem bistveno podaljšal časa nalaganja teme. Z uporabo tako izboljšanega algoritma računalniške animacije pričakujemo izboljšano konkurenčnost podjetja, saj sta vizualna privlačnost tem in kratek čas nalaganja za igralce dva pomembna dejavnika za uspešno trženje iger.

Zaključimo lahko, da smo z izboljšanim algoritmom računalniške animacije za igro slot dosegli zastavljene cilje. V nadaljevanju dela načrtujemo njegovo vgradnjo v ostale igre oz. aparate, ki jih izdeluje podjetje Gold Club. V prihodnje pa bi lahko tudi razširili uporabo animacij v stilu filma ali z uporabo ustreznega video kodeka s podporo prosojnosti ali s hranjenjem alfa kanala v sekundarnem filmu.

8 LITERATURA

Alpha compositing. Pridobljeno 21. 4. 2012 s svetovnega spleta:

http://en.wikipedia.org/wiki/Alpha_compositing

AVI Video textures. Pridobljeno 12. 5. 2012 s svetovnega spleta:

http://www.sulaco.co.za/opengl_project/AVI_video_textures.htm

Bin-Packing Problem. Pridobljeno 25. 3. 2012 s svetovnega spleta:

<http://mathworld.wolfram.com/Bin-PackingProblem.html>

DePetrillo, B. A. (2002). Razumeti Microsoft.NET. Ljubljana: Pasadena.

Erzetič, B., Gabrijelčič, H. (2010). 3D od točke do upodobitve. Ljubljana: Pasadena.

File system fragmentation. Pridobljeno 4. 4. 2012 s svetovnega spleta:

http://en.wikipedia.org/wiki/File_system_fragmentation

Frame rate. Pridobljeno 12. 5. 2012 s svetovnega spleta:

http://en.wikipedia.org/wiki/Frame_rate

Gold Club, About us. Pridobljeno 31. 5. 2012 s svetovnega spleta:

<http://www.gold-club.si/company/about-us>

Gold Club, Downloads, Brochure ENG. Pridobljeno 5. 2. 2013 s svetovnega spleta:

<http://www.gold-club.si/support/downloads>

GPU Memory Transfer. Pridobljeno 31. 5. 2012 s svetovnega spleta:

http://wiki.accelereyes.com/wiki/index.php?title=GPU_Memory_Transfer

HDD Charts 2012: Read Access Time: h2benchw 3.16. Pridobljeno 4. 4. 2012 s

svetovnega spleta: <http://www.tomshardware.com/charts/hdd-charts-2012/-07-Read-Access-Time-h2benchw-3.16,2898.html>

HDD Charts 2012: Read Throughput Average: h2benchw 3.16. Pridobljeno

31. 5. 2012 s svetovnega spleta: <http://www.tomshardware.com/charts/hdd-charts-2012/-01-Read-Throughput-Average-h2benchw-3.16,2901.html>

JPEG. Pridobljeno 12.5.2012 s svetovnega spleta: <http://en.wikipedia.org/wiki/JPEG>

Kerman, P. (2002). Naučite se flash MX v 24 urah. Ljubljana: Pasadena.

Lodi, A., Martello, S., Monaci, M. (2002). Two-dimensional packing problems:A survey. Pridobljeno 25. 3. 2012 s svetovnega spleta: <http://users.cs.fiu.edu/~yzhan004/rectangularPacking/survey.pdf>

Lossy compression. Pridobljeno 24. 4. 2012 s svetovnega spleta: http://en.wikipedia.org/wiki/Lossy_compression

Packing lightmaps. Pridobljeno 25. 3. 2012 s svetovnega spleta: <http://www.blackpawn.com/texts/lightmaps/>

PAL. Pridobljeno 21. 4. 2012 s svetovnega spleta: <http://en.wikipedia.org/wiki/Pal>

Patmore, C. (2003). The Complete Animation Course: The Principles, Practice and Techniques of Successful Animation. London: Thames & Hudson.

Popular Video Codecs for File Exchange. Pridobljeno 12. 5. 2012 s svetovnega spleta: http://documentation.apple.com/en/motion/usermanual/chapter_B_section_3.html

Portable Network Graphics. Pridobljeno 12. 5. 2012 s svetovnega spleta: http://en.wikipedia.org/wiki/Portable_Network_Graphics

Raster graphics. Pridobljeno 11. 12. 2011 s svetovnega spleta: http://en.wikipedia.org/wiki/Raster_graphics

RGB color model. Pridobljeno 17. 11. 2011 s svetovnega spleta: <http://en.wikipedia.org/wiki/Rgb>

S3 Texture Compression. Pridobljeno 31. 5. 2012 s svetovnega spleta: http://www.opengl.org/wiki/S3_Texture_Compression

SATA Hard Drives Run Riot: Maxtor, Hitachi, Western Digital 250 GB. Pridobljeno 31. 5. 2012 s svetovnega spleta: <http://www.tomshardware.com/reviews/sata-hard-drives-run-riot,709-8.html>

Video codec. Pridobljeno 12. 5. 2012 s svetovnega spleta:
http://en.wikipedia.org/wiki/Video_codec