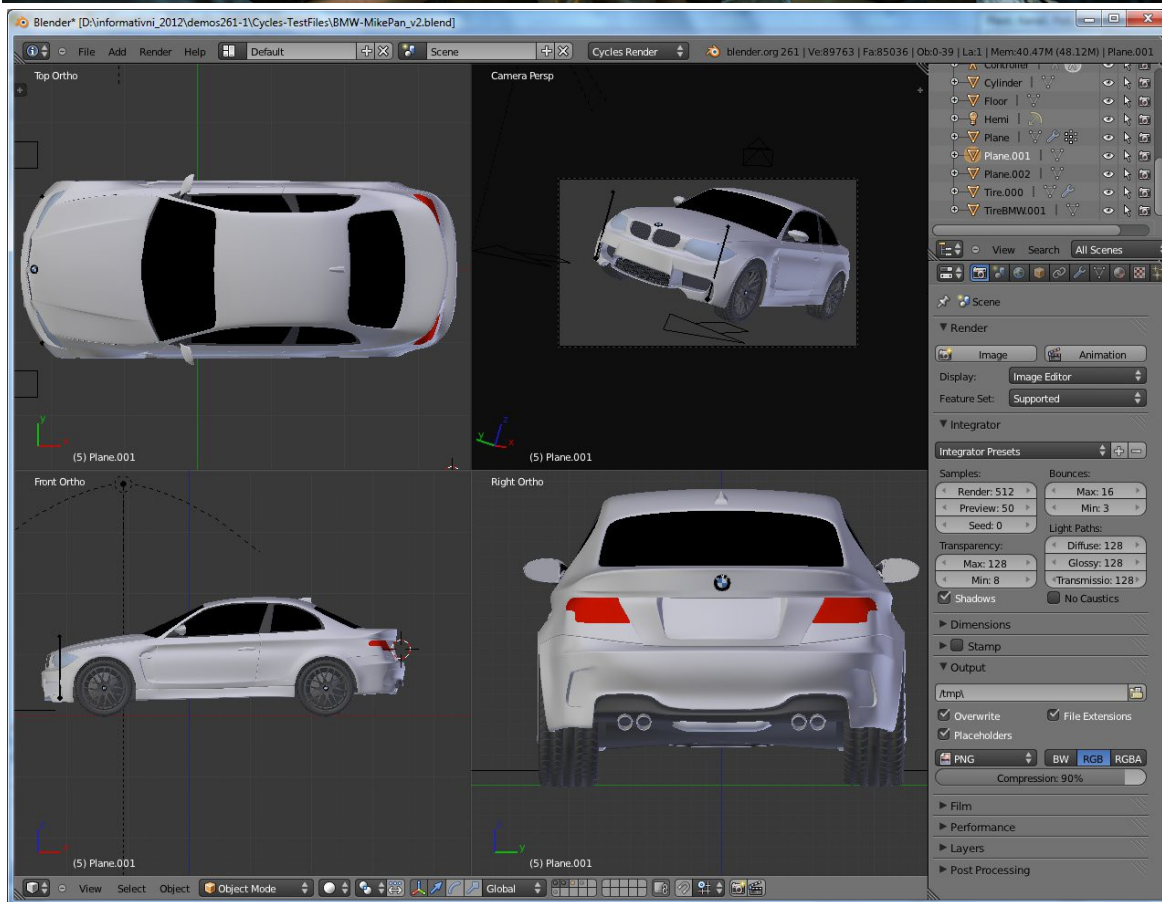


Uvod v 3D-animacijo s programom Blender 2.63



Avtor: Nedeljko Grabant

Velenje, avgust 2012

Kazalo vsebine:

1. Uvod v Blender 2.63
2. Učna situacija 1: Hiter začetek dela v Benderju z modeliranjem preproste mize
3. Učna situacija 2: Predstavitev uradnih projektov in izbranih izdelkov
4. Učna situacija 3: Od 3D-točke do upodobitve
5. Učna situacija 4: Lastnosti program Bledner
6. Učna situacija 5: Ustvarjanje napisov
7. Viri in literatura

Povzetek

Blender je odprtokodni program za ustvarjanje 3D-vsebin (slika 1), ki je na voljo za vse glavne operacijske sistema pod licenco GNU GPL (GNU General Public License). Na spletni strani <http://www.blender.org>, je tudi povezava (<http://www.blender.org/download/get-blender/>), s katere lahko brezplačno prenesete ta program na vaš računalnik. S programom Blender je možno modelirati (ustvarjati 3D-predmete ali modele), animirati, upodabljati (ustvariti filme ali slike), izvesti kompozicijo in filmsko montažo vključno z zvokom in 3D-interaktivno vsebino oz. igro (angleško je na kratko to model - shade - animate - render - composite - interactive – 3d). V tem gradivo bomo obdelali osnovno teorijo in praktične primere od modeliranja do končne filmske montaže. Namen je podati celoviti uvod v 3Danimacijo primerno za srednješolsko populacijo in ponekod je gradivo primerno tudi za višje in visoke šole.

Ključne besede: Blender, 3D, modeliranje, NURBS, Bezier, krivulje, animacija, film, video montaža

Avtorji: Nedeljko Grabant

Drugi avtorji (slikovno, multimedijsko gradivo ...): Nedeljko Grabant

Recenzent: Peter Rau

Lektor: Marija Klemenšek

Datum: November 2011

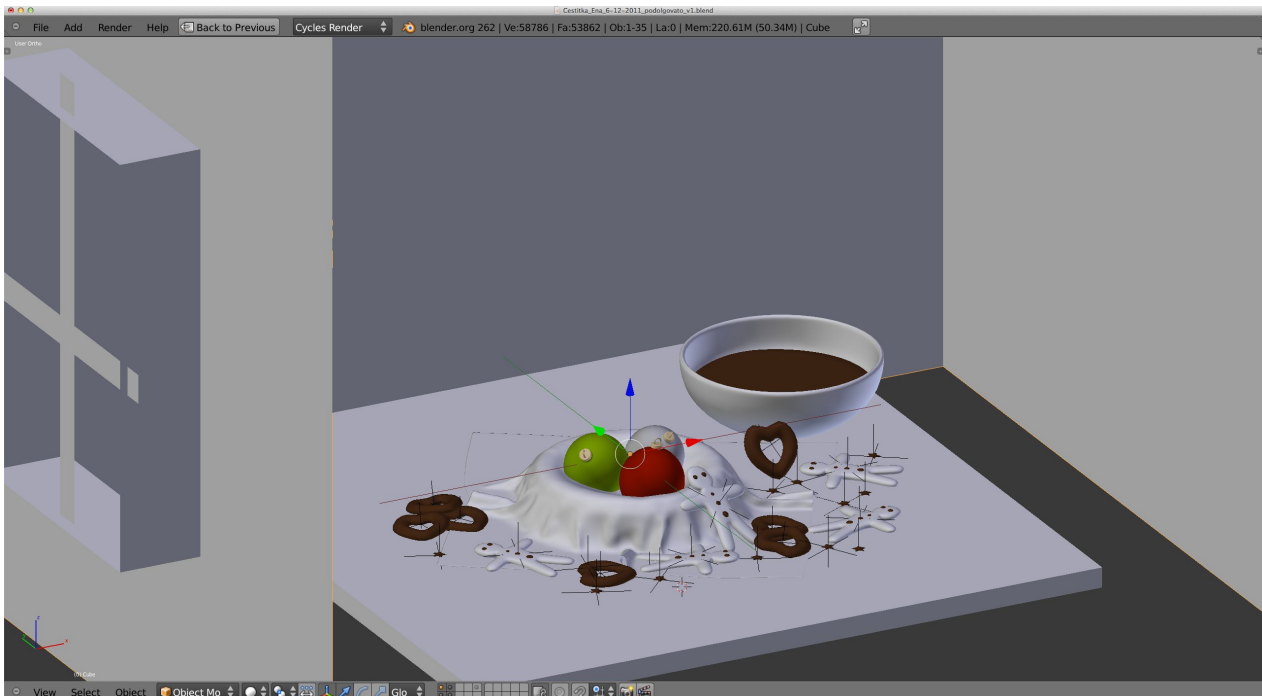
CIP – Kataložni zapis o publikaciji (poskrbi koordinator)



To delo je ponujeno pod Creative Commons Priznanje avtorstva-Nekomercialno-Deljenje pod enakimi pogoji 2.5 Slovenija licenco.

1 Uvod v 3D-animacijo s programom Blender 2.63

Blender je odprtokodni program za ustvarjanje 3D-vsebin (slika 1), ki je na voljo za vse glavne operacijske sistema pod licenco GNU GPL (GNU General Public License). Na spletni strani <http://www.blender.org>, 10. 02. 2012, je zapisana angleška definicija »Blender is the free open source 3D content creation suite, available for all major operating systems under the [GNU General Public License](http://www.gnu.org/licenses/gpl.html).« Na tej strani je tudi povezava (<http://www.blender.org/download/get-blender/>), s katere lahko brezplačno prenesete ta program na vaš računalnik. S programom Blender je možno modelirati (ustvarjati 3D-predmete ali modele), animirati, upodabljati (ustvariti filme ali slike), izvesti kompozicijo in filmsko montažo vključno z zvokom in 3D-interaktivno vsebino oz. igro (angleško je na kratko to model - shade - animate - render - composite - interactive - 3d).



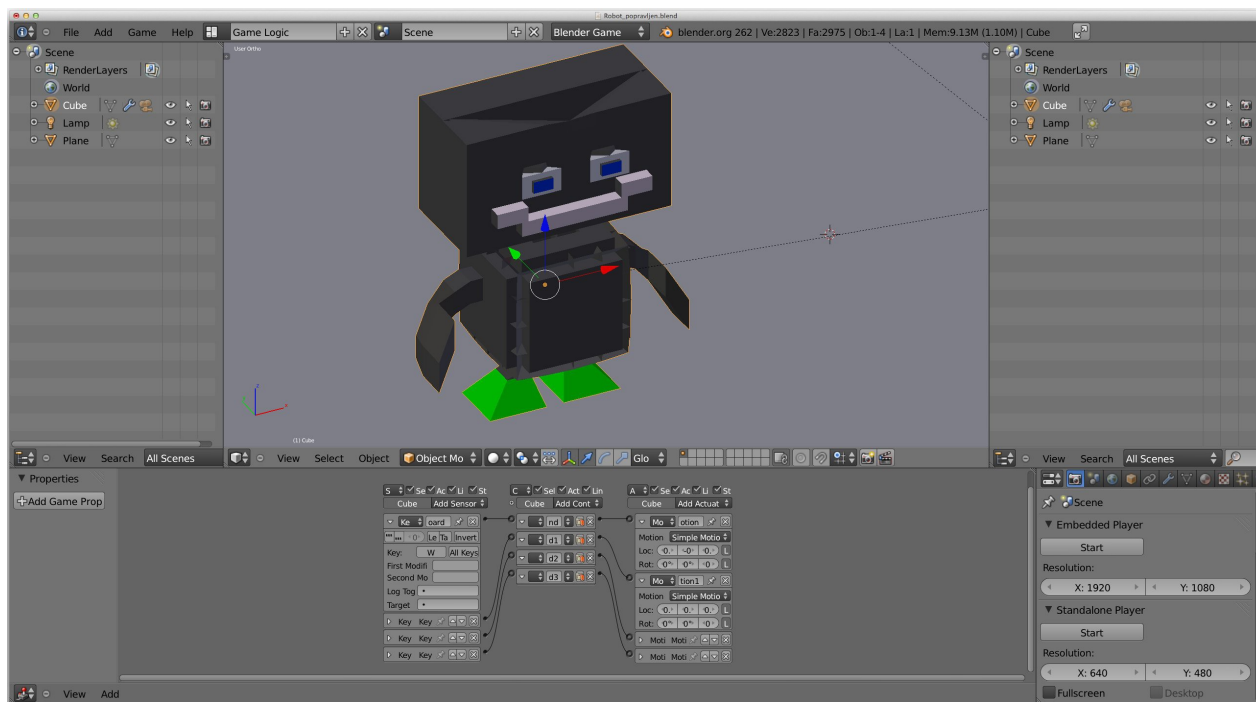
Slika 1: Primer 3D-scene, pripravljen za upodabljanje, avtor scene je Blaž Pečnik

Beseda Blender izvira iz angleške besede Blend, ki pomeni (z)mešati, spojiti, (z)družiti, zli(va)ti, preli(va)ti se (tako se v angleščini imenuje tudi mešalnik in sekljalnik za hrano). Torej slike (slika 2) nastajajo pri upodabljanju, pri animaciji se prelivajo in nastane gibanje ali filmski zapis.






Slika 2: Primer upodobljene slike iz prve slike primera 3D-scene, avtorja Blaža Pečnika


Blender ima številne lastnosti. Ena od najpomembnejša lastnosti je, da je popolnoma prosto in odprtokodno 3D-orodje za uporabo in razširjanje (distribucijo) v učne, profesionalne (poslovne) ali komercialne namene uporabe. Umetniški izdelki, ki nastanejo z Blenderjem, vključno s Pythonovimi skriptami, so le v lastništvu ustvarjalca ali ustvarjalcev.

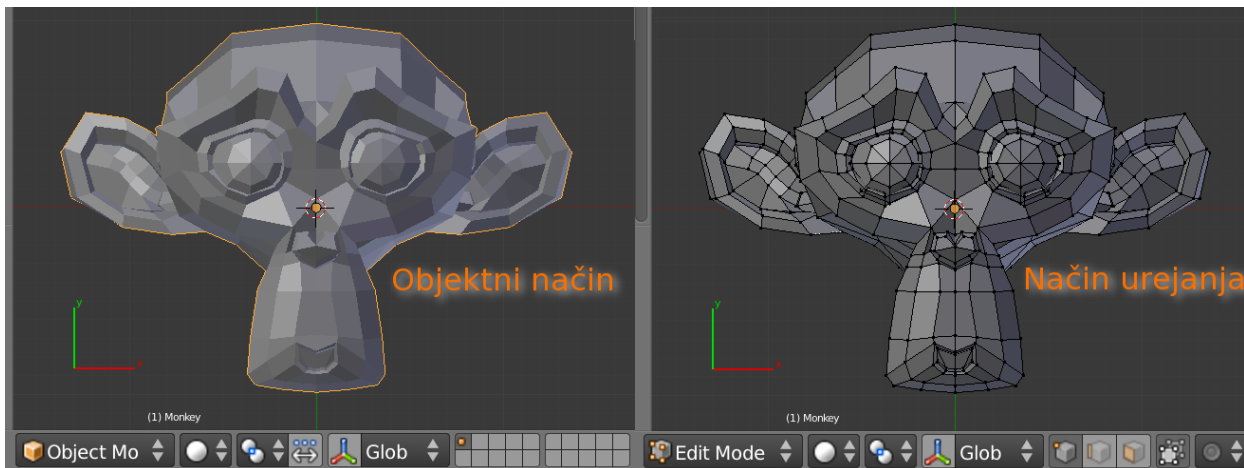


Slika 3: Primer Blenderjevega ustvarjanja 3D-igrice, avtor Primož Pesjak


Za bolj učinkovito in hitro delo z Blenderjem je smiselno imeti eno roko na tipkovnici (takšna, ki ima tudi numerično tipkovnico) in drugo na miški. Miškini simboli , ,  označujejo pritisnjeno levo, sredinsko in desno tipko miške.

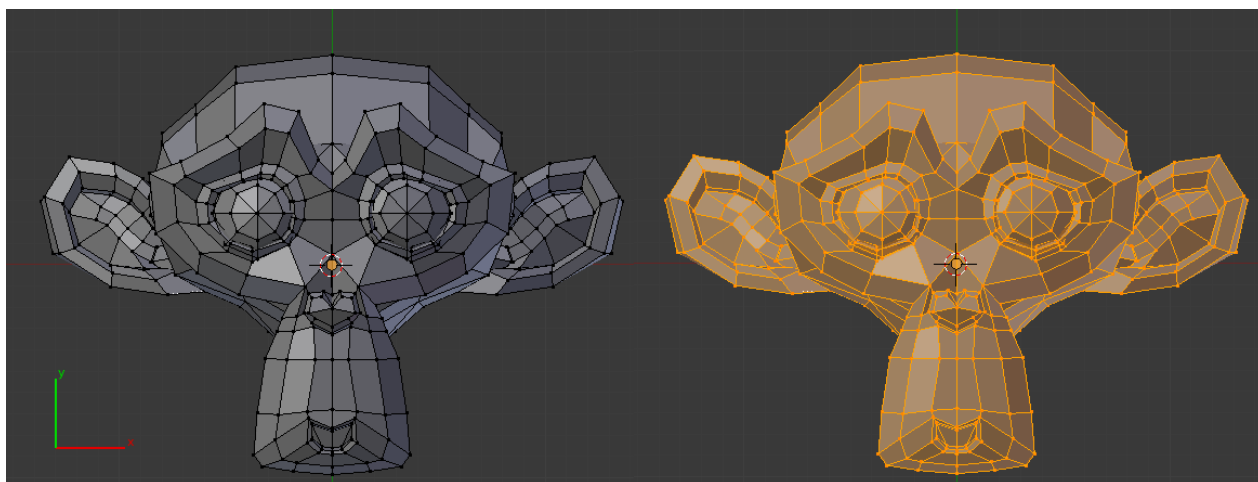
Za zapise bližnjic je v teh navodilih uporabljena posebna odprtokodna pisava Linux Biolinum Keyboard (če bi jo kdo rad uporabljal, jo lahko najde na spletu in jo je možno namestiti na npr.

operacijskem sistemu Windows v mapo C:\Windows\Fonts). V njej je npr. za tabulatorska tipka označena kot , tako kot je to na tipkovnici in pomeni preklop iz Objektnega načina (angl. Object Mode) v Način urejanja (angl. Edit Mode) ali obratno (slika 4).



Slika 4: S pritiskom na tipko TAB priklapljamemo med Objektim načinom in Načinom urejanja

Podobno je npr. črka , ki je bližnjica za Izberi vse (angl. Select All). Z njenim ponovnim pritiskom se vklopi funkcija Izberi nič (angl. Deselect All) (vse tipke imajo to dvojno preklopno funkcijo. Z enim pritiskom vklopimo funkcijo in z drugim pritiskom izklop funkcije - slika 5).



Slika 5: Ničesar izbranega (levo) in vse izbrano (desno)

V nadaljevanju bo najprej skozi primer modeliranja pravokotne mize opisana prva uporaba Blenderja. Pozneje se bomo bolj podrobno posvetili lastnostim ter skozi praktične primere spoznali posamezne funkcije Blenderja.



Vprašanja za preverjanje razumevanja poglavja 1 Uvod

4

Učno gradivo je nastalo v okviru projekta Munus 2. Njegovo izdajo je omogočilo sofinanciranje Evropskega socialnega sklada Evropske unije in Ministrstva za šolstvo in šport.

Pisno in/ali ustno odgovori na vprašanja v nadaljevanju.

Kaj pomeni beseda Blender in od kod prihaja?

Na spletni strani <http://www.blender.org/> poišči zanjo aktualno različico programa Blender (številko verzije)? Kdaj je bila ta objavljena in ali se kje dobijo tudi prejšnje verzije tega programa?

Naštej vse možnosti, za katere se lahko uporabi program Blender!

Kateri skriptni jezik uporablja Blender?

Kdo ima avtorske pravice oz. lastništvo nad umetniški izdelki, ki nastanejo z Blenderjem vključno s Pythonovimi skriptami?

Kako se najbolj učinkovito in hitro dela z Blenderjem (kje so roke)?

Napiši pot do mape, kje se v operacijskem sistemu Windows nahaja mapa s pisavami!

Napiši pot do mape, kje se v operacijskem sistemu Mac in Linux nahaja mapa s pisavami!

S katero bližnjico se preklopi med Objektnim načinom (angl. Object Mode) in Načinom urejanja (angl. Edit Mode)?

S katero bližnjico vklopi funkcija Izberi vse (angl. Select All) za vse objekte ali oglišča?



Naloge za poglobljanje poglavja 1 Uvod

Odgovore na naslednja vprašanja je možno dobiti na Blenderjevi wikipediji (<http://wiki.blender.org/index.php/Doc:2.6/Reference/>).

Preglej Blenderjevo wikipedijo in v angleščini in slovenskem prevodu naštej posamezna glavna poglavja te wikipedije!

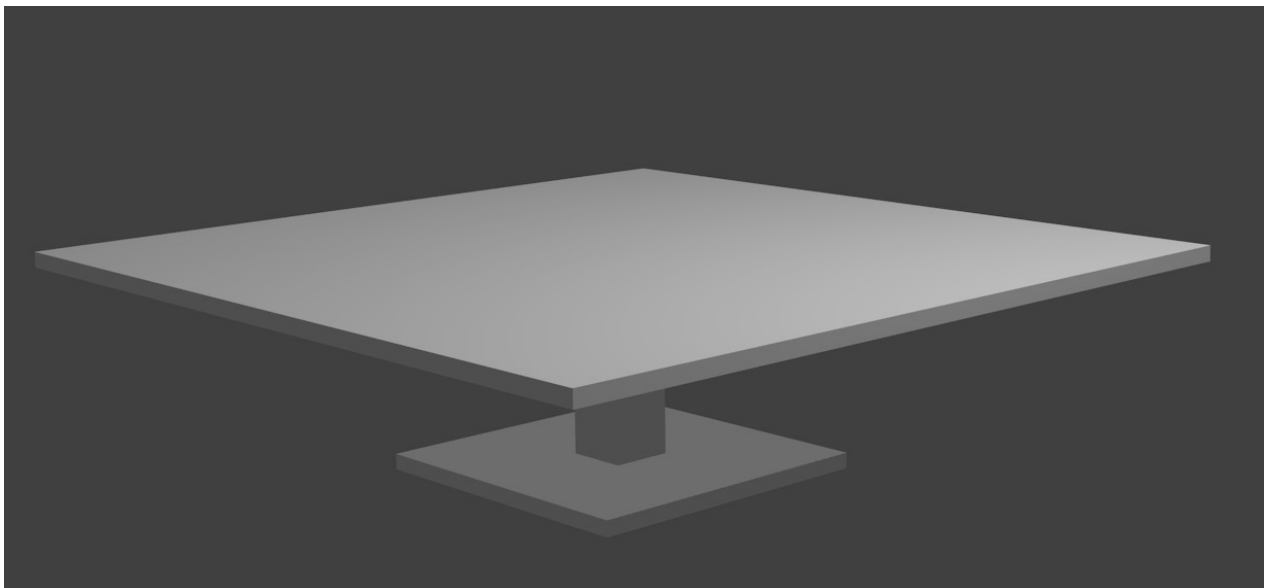
Za katero poglavje oz. področje v Blenderjevi wikipediji meniš, da je najboljše predstavljeno?

Kje se dobijo proste knjige o Blenderju?

Učna situacija 1: Hiter začetek dela v Blenderju na primeru modeliranja preproste mize

Namen tega poglavja je začetnikom ponuditi krajše navodilo za delo z Blenderjem. Namenjeno je vsem tistim, ki bi radi na hitro nekaj naredili v 3D-prostoru pa nimajo začetnega znanja o uporabi programa.

Izbran je primer modeliranja
modeliranja preproste mize (slika 6).



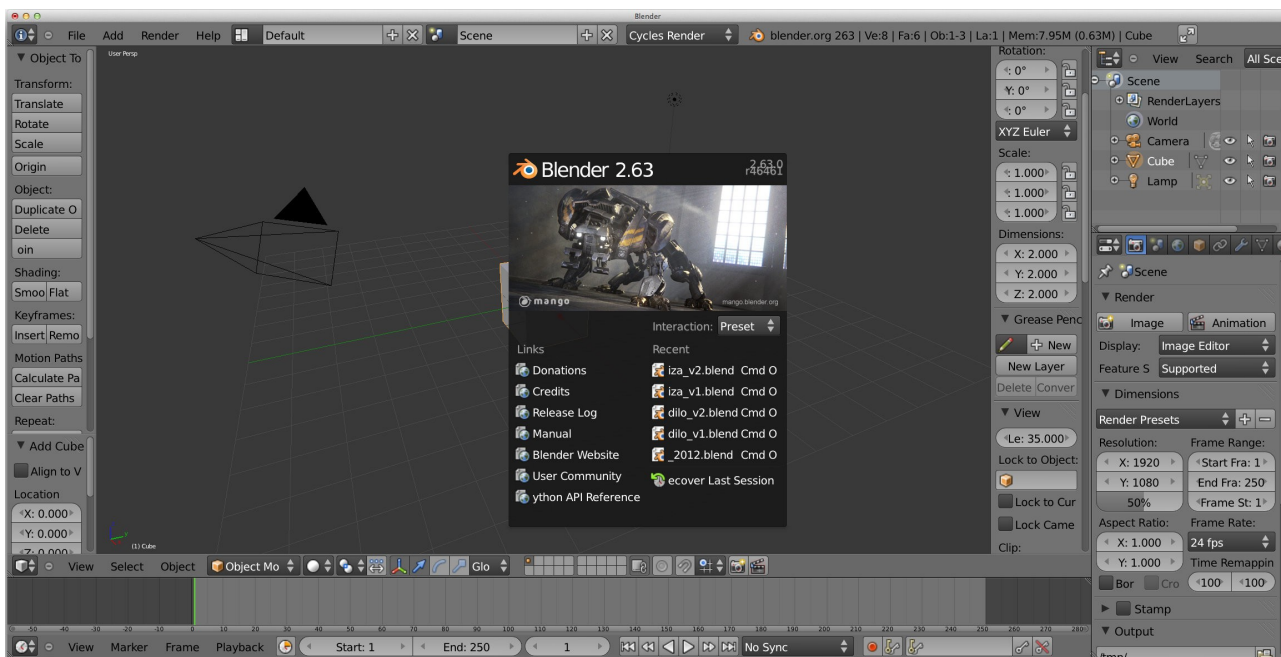
Slika 6: Končni izgled mize

Predpogoj za začetek dela je nameščen program Blender, ki ga lahko brezplačno prenesete s spletne povezave <http://www.blender.org/download/get-blender/> (slika 7).

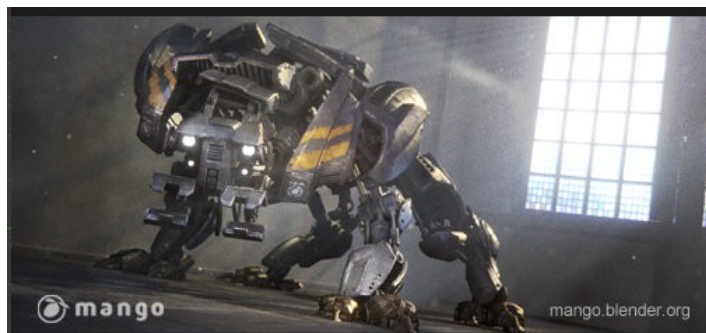
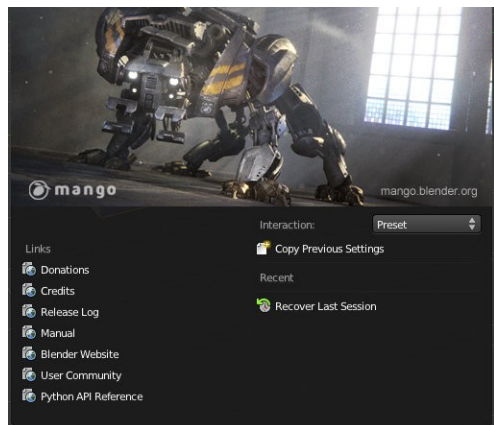


Slika 7: Blender lahko prenesete s klikom na ikono z napisom Download Blender

Blender namestite z dvojnimi klikom na preneseno datoteko in nato v pogovornem oknu izberete gumb naprej (angl. Next) oz. naprej. Po nameščanju lahko program zaženete z namizja tako, da z dvojnimi klikom izberete ikono Blender. Na zaslonu se pojavi programsko okno Blenderja s pozdravno sliko (angl. Splash), na kateri je vidna številka različice programa in desno zgoraj je serijska številka izdaje (slika 8).



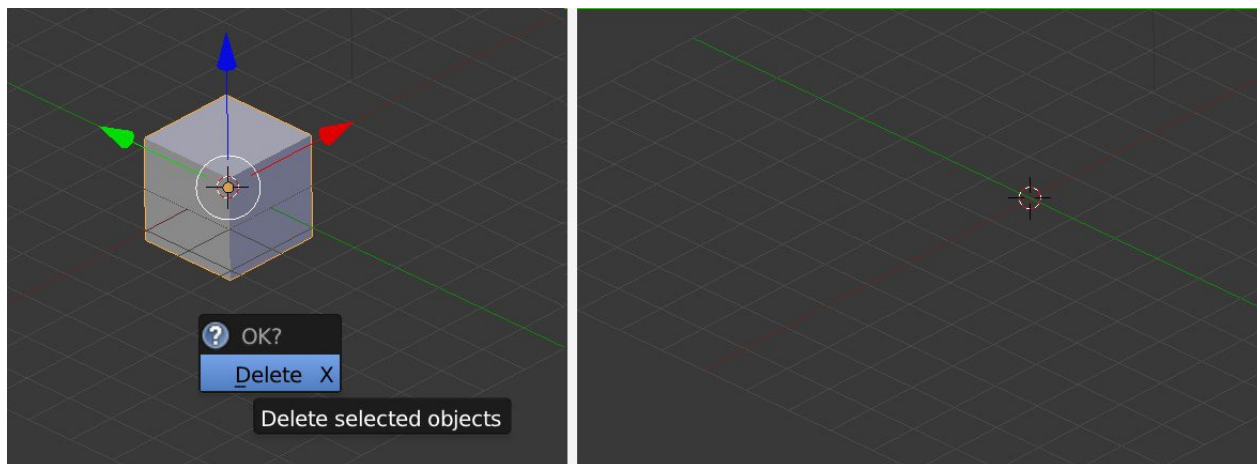
Slika 8: Prvi pogled na program po zagonu Blenderja 2.63 na računalniku z OS MAC



Slika 9: Zagonska ali angl. Splash slika Blenderja (levo s povezavami in nazadnje odprtimi datotekami)

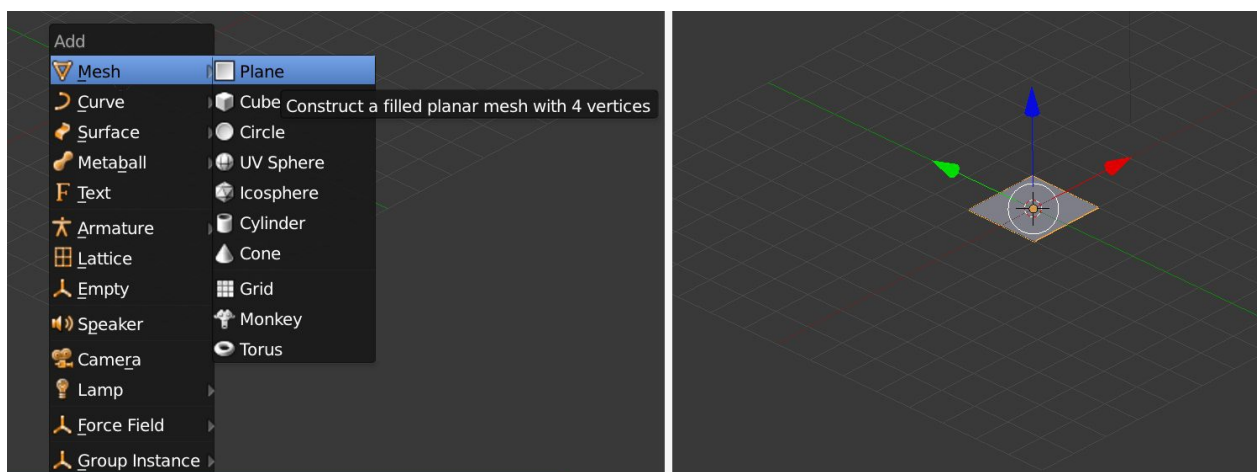
Po kliku v sredinsko okno t. i. okno 3D-pogled izgine ta grafika s svetlečo rožo, ki vsebuje napis različice Blenderja 2.63 (slika 9). Pri modeliranju je kot pri vsaki spretnosti smiselno vaditi in utrjevati obstoječe spretnosti. V primeru, da se nam pri delu kaj ponesreči, lahko enkrat ali večkrat uporabimo bližnjico za razveljavitev **[Ctrl]** - **[Z]**.

Na začetku izbrišemo obstoječo kocko tako, da pritisnemo tipko **[X]** in na zaslonu se pojavi opozorilo o brisanju, ki ga podredimo s tipko (slika 10). V koordinatnem središču je viden 3D-kazalec (angl. 3D Cursor).





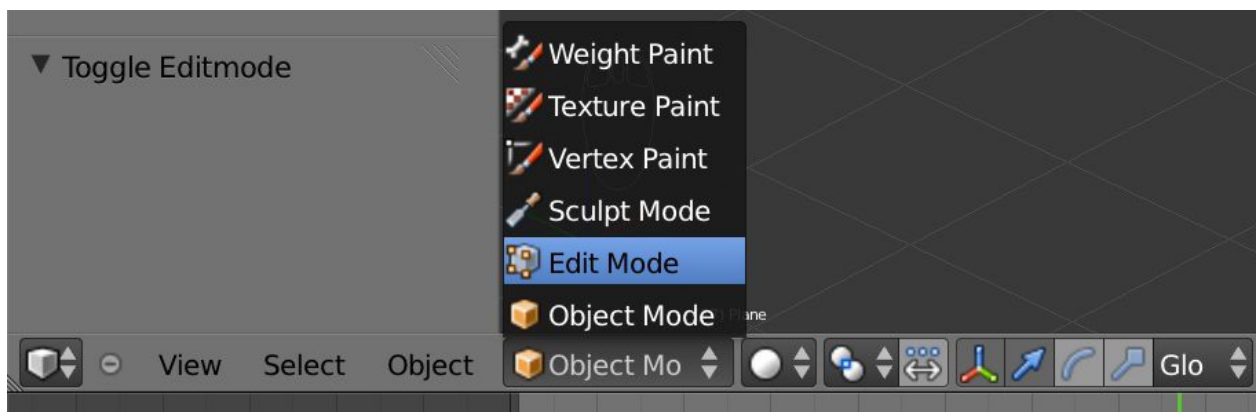
Slika 10: Pred brisanjem kocke in po njem

Vstavimo ravnino angl. Plain, ki bo osnova za nadaljnje modeliranje mize. Ta bo vstavljena v koordinatno središče tam, kjer je viden 3D-kazalec (to je krog iz rdečih in belih lokih ter navpičnima in vodoravnima črnima črtama desna slika 10) tako, da pritisnemo kombinacijo tipk **[⇧]** - **[A]** (torej Shift + A). V priročnem meniju izberemo pod Mesh – Plane (Večkotniška mreža -Ravnina), ki je oznaka za ravnino. To krajše zapišemo kot Add | Mesh | Plane (leva slika 11). Na položaju 3D-kazalca se pojavi vstavljena ravnina (desno na sliki 11).





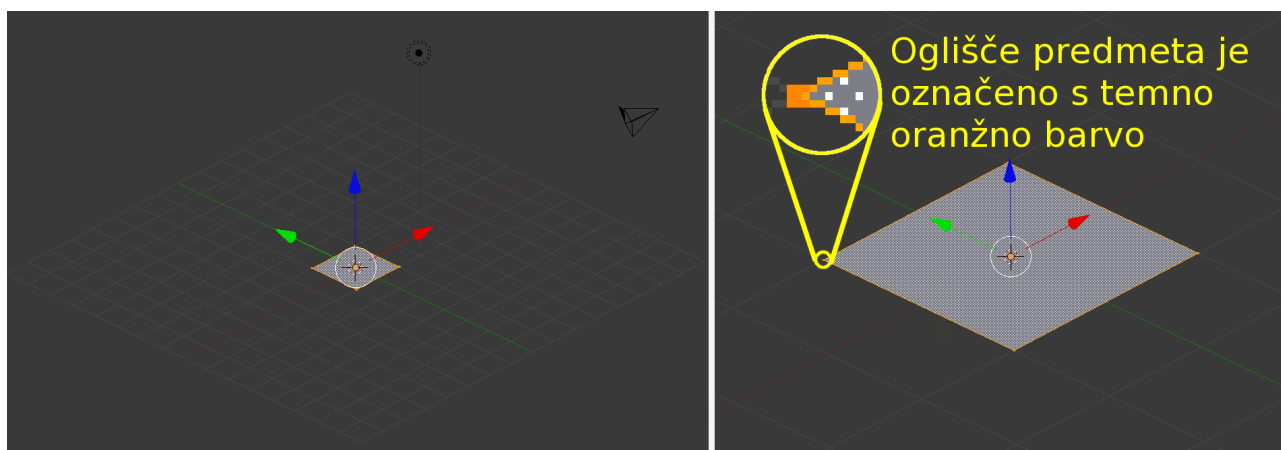
Slika 11: Meni za vstavljanje ravnine Add | Mesh | Plain (levo na sliki) in po vstavljanju (desno na sliki)

S pritiskom na tipko TAB preklapimo iz Objektnega načina (angl. Object Mode) v Način urejanja oz. (angl. Edit Mode), kjer lahko oblikujemo lokalni izgled objekta po ogliščih. Na tipkovnici je tipka TAB označena z . Preklop izvedemo lahko tudi s klikom na izbirni meni način, ki je v spodnji glavi okna 3D-pogled in tam izberemo Način urejanja oz. (angl. Edit Mode) (slika 12). Seveda je uporaba tipke  dosti hitrejši način preklopa v Način urejanja objekta s pomočjo oglišč in jo je zato smiselno čim več uporabljati, da nam ta bližnjica postane samoumevna oz. gre v nezavedno uporabo ali avtomatizem.



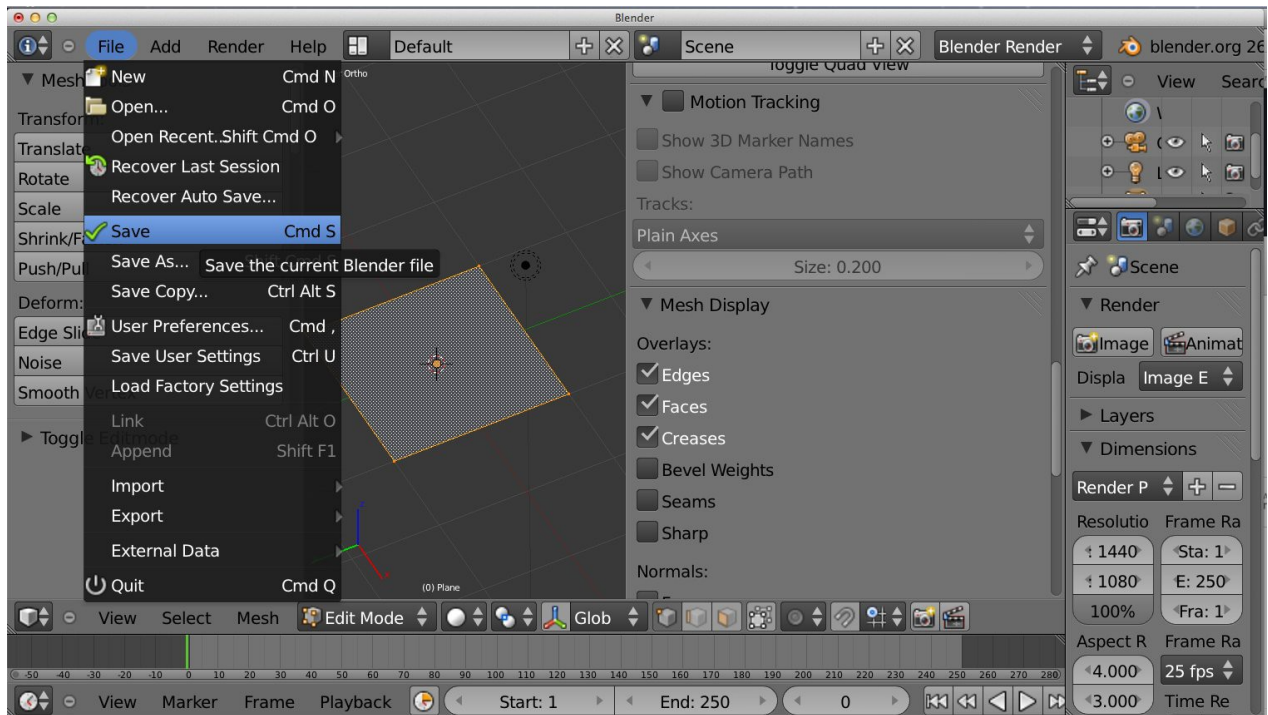
Slika 12: Preklop v način urejanja oz. Edit Mode

Sredinsko okno 3D-pogled (angl. 3D View) je glavno delovno okno za modeliranje. Za boljši pogled na sceno je smiselno povečati predmet oz. sceno. To najlažje storimo z vrtenjem kolesčka miške  ali z uporabo tipk **Pad+** za večanje in tipke **Pad-** za manjšanje, ki sta na numerični tipkovnici. Z večanjem predmeta se večja tudi navidezna mreža, ki je v ozadju predmeta (slika 13) in ta omogoča primerjavo in lažje modeliranje predmetov. Pri vrtenju kolesčka miške  od sebe večamo razdaljo opazovanja na predmet, pri obratnem sukanju k sebi pa zmanjšujemo zmanjšujemo razdaljo opazovanja zato sta mreža scene (prizorišča) in predmet videti večja (slika 13).



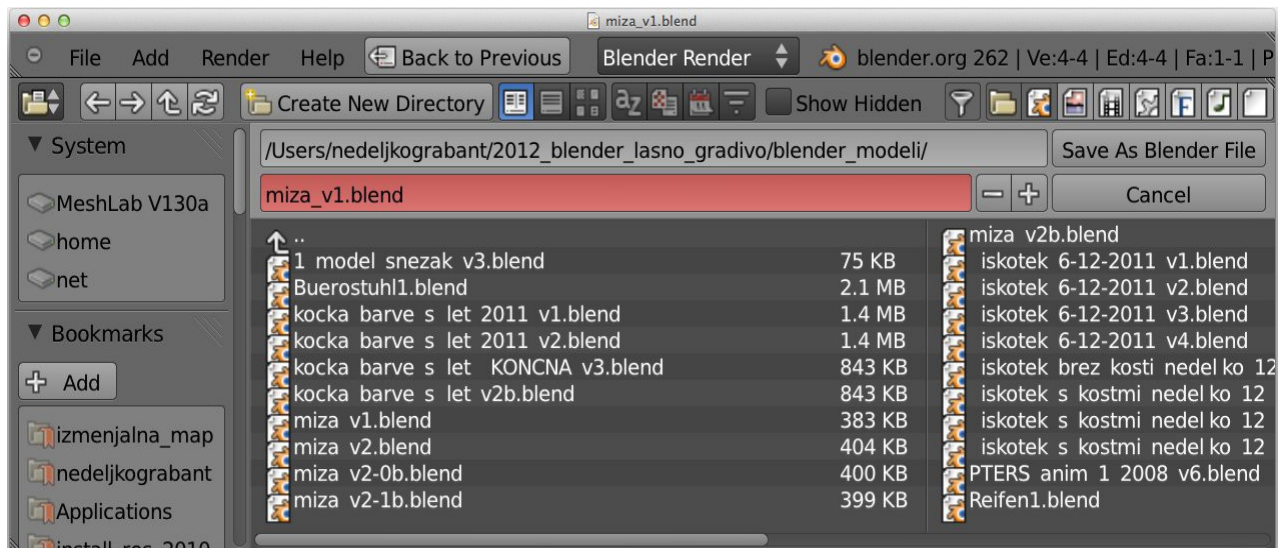
Slika 13: Pred in po večanju pogleda z vrtenjem kolesčka miške  ali z uporabo tipk **Pad+** in **Pad-**

Shranimo trenutno stanje modela mize v datoteko, ki ima v programu Blender končnico **blend** (pozneje lahko z odpiranjem te shranjene datoteke preprosto nadaljujemo z delom ali s spremembami na objektu in v sceni (prizorišču)). Za shranjevanje lahko uporabimo tudi bližnjico **Ctrl** - **S** (na MAC OS X je prava tipka **Cmd** oz. **⌘** - **S**) ali hitreje s pritiskom na funkcijsko tipko **F2**. Tretja možnost shranjevanja je z uporabo menijske vrstice, v kateri izberemo ukaz File | Save oz. Datoteka | Shrani (slika 14).




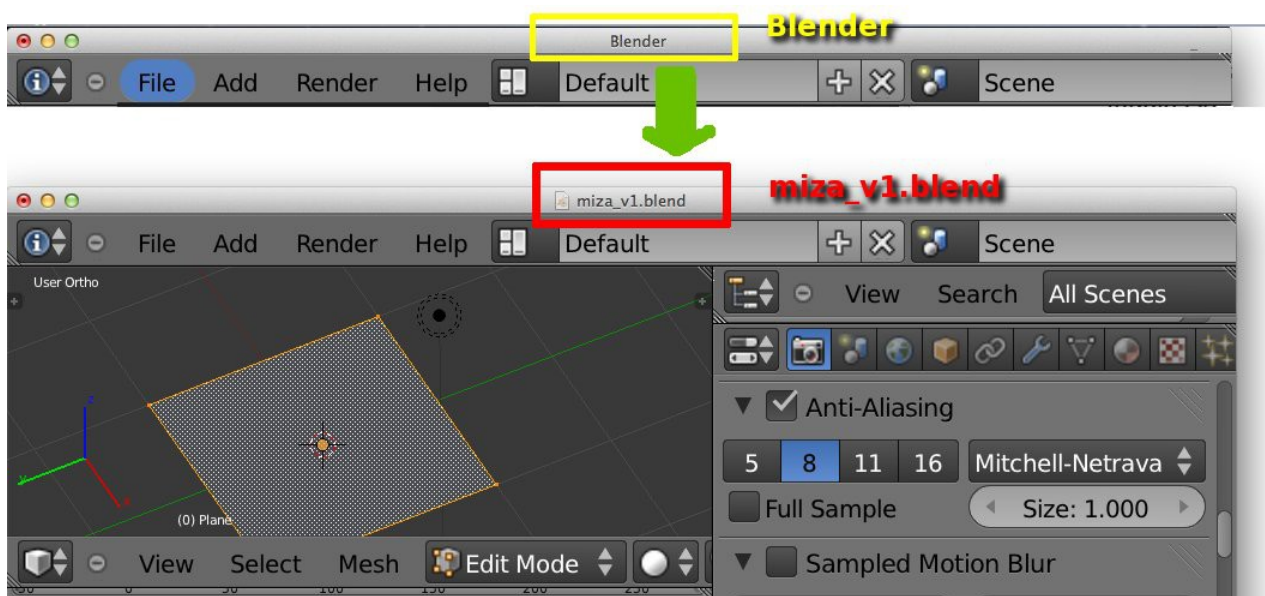
Slika 14: Shranjevanje trenutnega stanja modeliranja mize z uporabo menija ali hitreje to dosežemo z bližnjico **Ctrl** - **S** oz. kar s funkcijsko tipko **F2**

Na zaslonu se odpre pogovorno okno, kjer lahko izberete mesto shranjevanja in v vnosno polje levo od znaka **-** vpišete ime datoteke, npr. miza_v1.blend (slika 15).



Slika 15: Pogovorno okno za vnos imena datoteke Blender za nastajajočo mizo z imenom *miza_v1.blend*

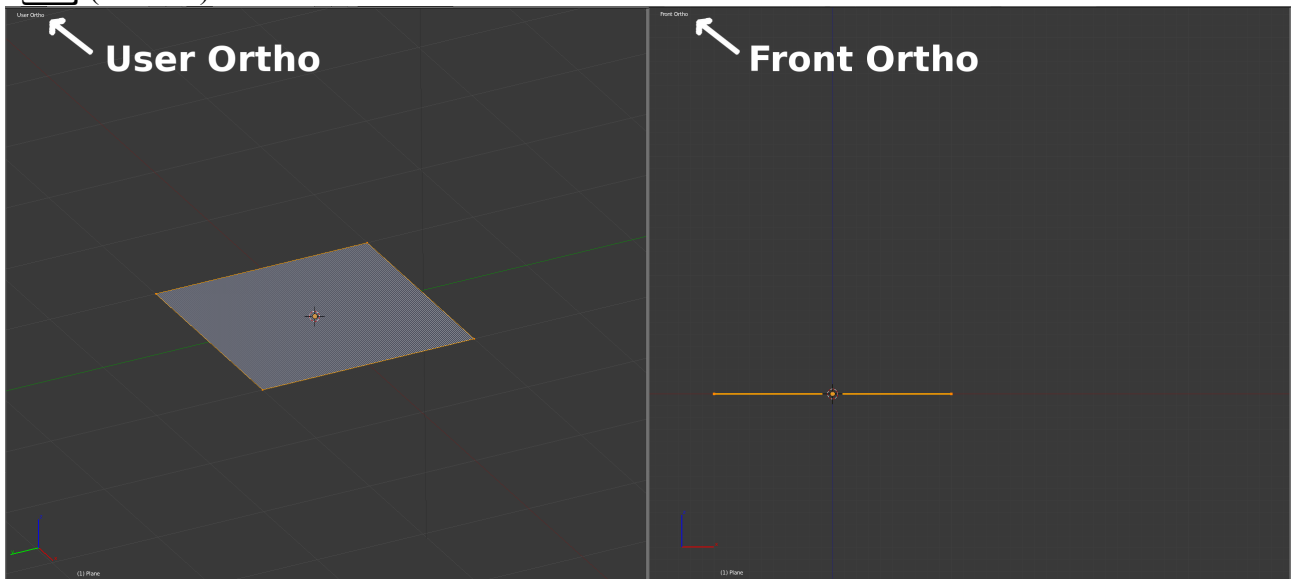
Po dvojnem pritisku tipke  ali s klikom na gumb [Save As Blender File] se zapiše nastajajoča datoteka Blender na izbrano mesto na podatkovnem nosilcu, ki je v našem primeru trdi disk. Po uspešnem shranjevanju Blenderjeve datoteke se vidi sprememba imena v naslovni pasici programa iz imena **Blender** v ime datoteke npr. **miza_v1.blend** (slika 16).




Slika 16: Po shranjevanju se spremeni ime v naslovni pasici programa iz Blender v *miza_v1.blend*

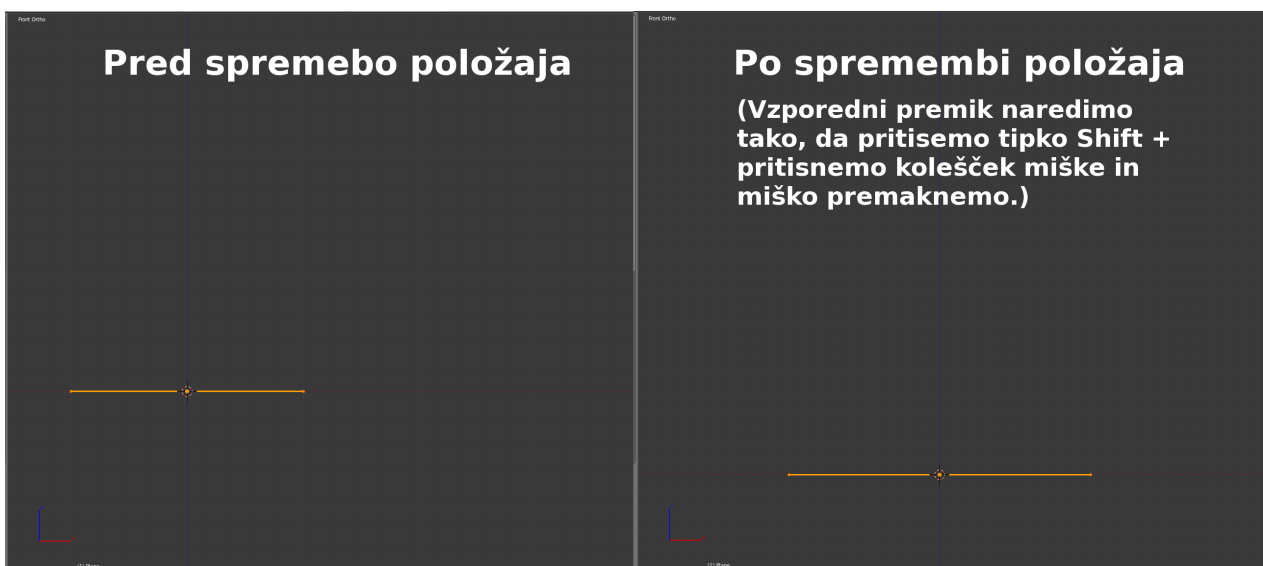
Sedaj spremenimo pogled iz perspektivnega v ortogonalni ali pravokotni z uporabo numerične tipke **[Pad5]**, ki je bolj primeren za modeliranje. V zgornjem levem vogalu okna 3D-pogled je napisano User Ortho (pravokotni pogled). Zamenjajmo pogled v naris z uporabo numerične tipke

Pad 1 (slika 17).




Slika 17: Pravokotni pogled in sprememba pogleda iz uporabniškega v pravokotni oz. naris

Za lažje delo pri modeliranju vzporedno spremenimo trenutni položaj pogleda. Vzporedni premik ali translacijo pogleda v oknu 3D-pogled naredimo tako, da pritisnemo tipko Shift (pravilno slovensko je dvigovalka vendar zaradi kajšega zapisa in ker so uporabniki bolj navajeni) oz.  + in držimo pritisnjen kolesčček miške ter miško premaknemo v želeno smer spremembe položaja (slika 18). S tem poskusimo premakniti sliko tako, da v spodnjem in vodoravno sredinskem delu premaknemo pogled na spodnjo nogo bodoče mize.



Slika 18: Pred spreminjanjem položaja objekta v oknu 3D-pogled in po njem

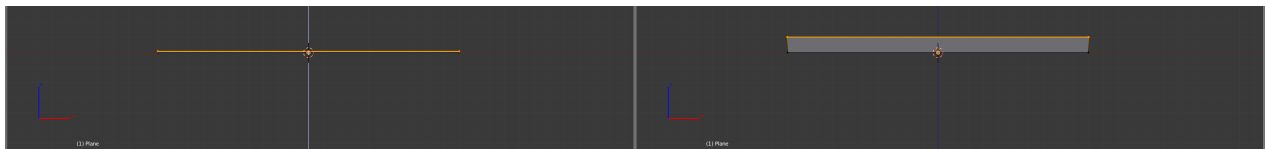
S pogledom v spodnjo vrstico preverite, če ste v Način urejanja (angl. Edit Mode), če tam piše Object Mode pritisnite tipko TAB oz. , ker sicer ne morete izvajati lokalne operacije nad

12

Učno gradivo je nastalo v okviru projekta Munus 2. Njegovo izdajo je omogočilo sofinanciranje Evropskega socialnega sklada Evropske unije in Ministrstva za šolstvo in šport.

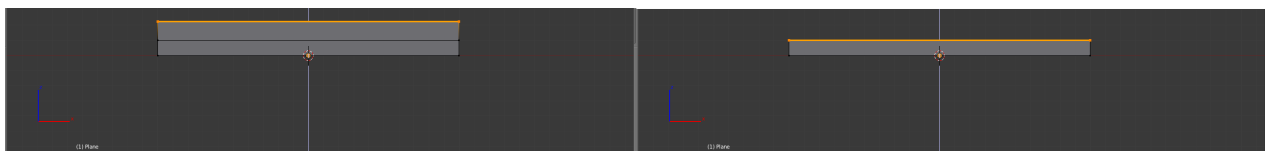
izbranimi oglišči predmeta.

Izvečemo oglišča (angl. Extrude Vertices) tako, da pritisnemo tipko **[E]**, potem pritisnemo še tipko **[Ctrl]** in premaknemo miško navzgor tako, da omejimo gibanje npr. za eno enoto (to je debelina stojala mize). Pri tem se na zaslonu izriše modra črta za z-os, po kateri lahko delamo izvlek (slika 19).



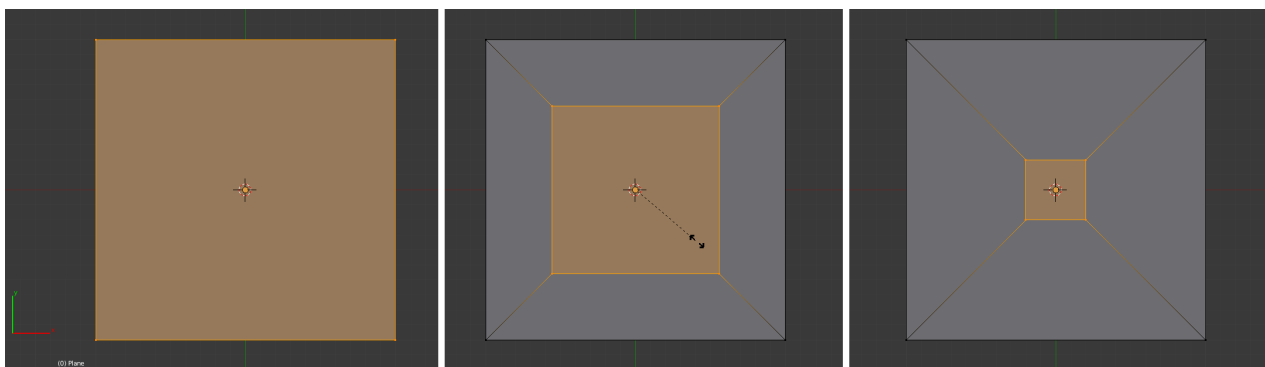
Slika 19: Pred izvlekom (bližnjica **[E]**) "stojala" za nogo bodoče mize in po njem

Za nogo mize naredimo še en izvlek z bližnjico **[E]** (angl. Extrude) in prekinemo operacijo premika s kratkim pritiskom na tipko **[Esc]**, ki je samodejno vklopljena po vklopu izvleka (slika 20).



Slika 20: Pred izvlekom in po preklicu premika izvleka s tipko **[Esc]**

Prestavimo se v pogled v tloris z bližnjico **[Pad 7]** (stalno imamo vklopljen numerični del tipkovnice). V ta pogled smo se prestavili zato, ker bi radi zmanjšali presek noge mize tako, da to vidimo iz samega pogleda od zgoraj (leva slika 21). Za spremembo debeline noge enkrat pritisnemo na tipko **[S]** (to je skrajšano od besede skaliranje (angl. Scale) in pomeni večanje ali manjšanje) in zraven držimo tipko **[Ctrl]** tako, da se ob premikanju miške proti notranjosti podnožja spreminja debelina noge po velikosti mreže npr. do dveh enot (desna slika 21).

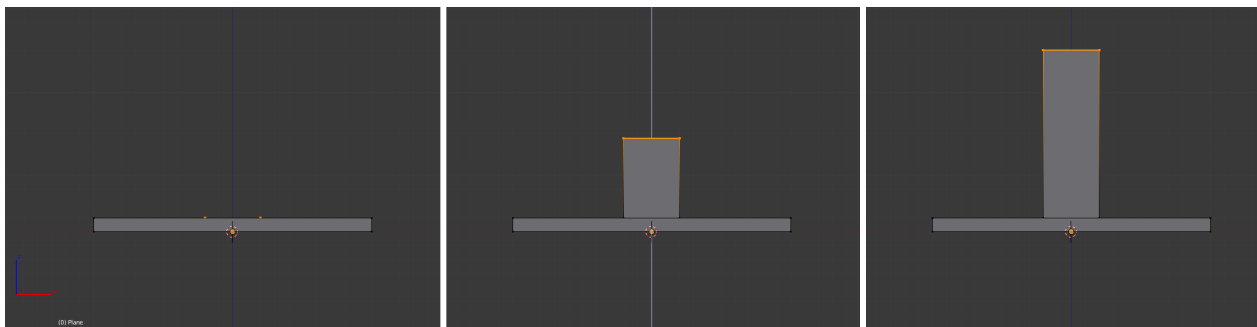


Slika 21: Pred skaliranjem preseka noge mize, med njim in po njem

Zamenjajmo pogled v naris z uporabo numerične tipke **[Pad 1]** (levo na sliki 22). Izvečemo oglišča (angl. Extrude Vertices) tako, da pritisnemo tipko **[E]**, potem pritisnemo še tipko **[Ctrl]** in premaknemo miško navzgor tako, da omejimo gibanje oglišč npr. za 13 enot (to je višina noge do 13

Učno gradivo je nastalo v okviru projekta Munus 2. Njegovo izdajo je omogočilo sofinanciranje Evropskega socialnega sklada Evropske unije in Ministrstva za šolstvo in šport.

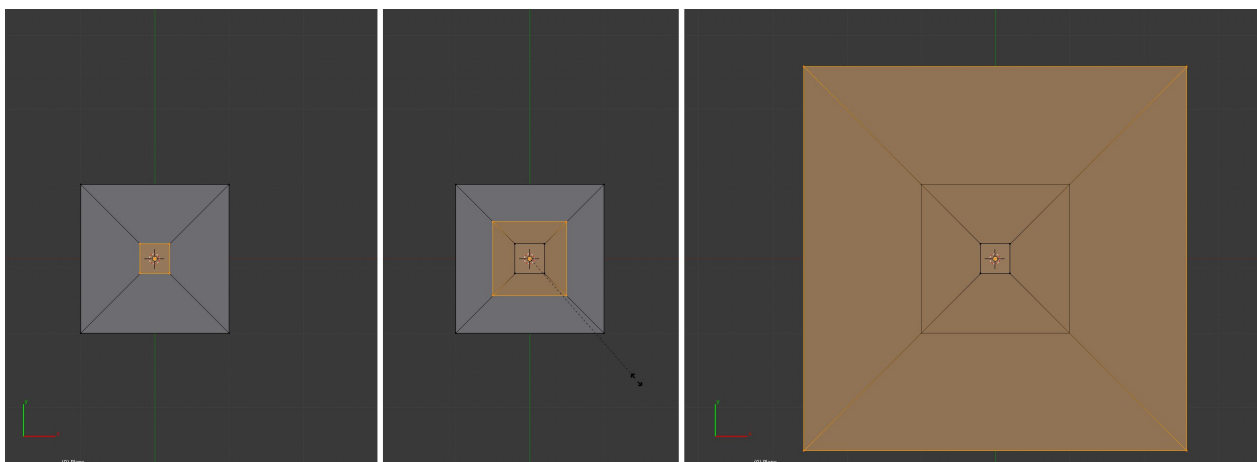
zgornje plošče mize). Pri tem se na zaslonu izriše na zaslonu modra črta, z-os, po kateri lahko delamo premik po izvleku (srednja slika 22).



Slika 22: Izvlek oglišča s pritiskom tipke **[E]** in tipke **[Ctrl]** ter premik miške navzgor npr. za 13 enot

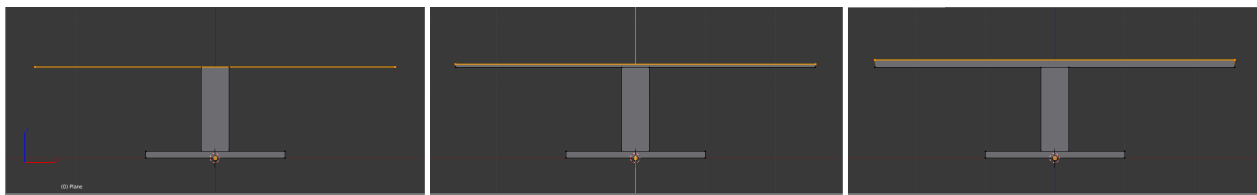
Modelirajmo še zgornjo ploščo mize tako, da naredimo še en izvlek z bližnjico **[E]** in prekinemo operacijo premika s pritiskom na tipko **[Esc]**.

V pogled tloris se prestavimo z bližnjico **[Pad 7]** tako, da bi lahko videli spremembo velikosti zgornje plošče (leva slika 23). Za ustvarjanje dimenzije zgornje plošče mize bomo izbrane točke skalirali z enkratnim pritiskom na tipko **[S]** in zraven držali tipko **[Ctrl]** tako, da se ob premikanju miške navzven spreminja dimenzija plošče po enotah mreže do zelene velikosti (desna slika 23).



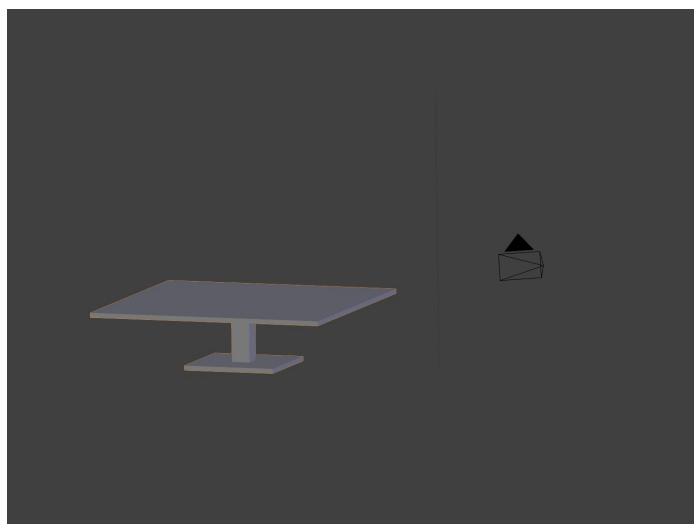
Slika 23: Pred skaliranjem zgornje plošče mize, med njim in po njem

Zamenjajmo pogled v naris z uporabo numerične tipke **[Pad 1]** (levo na sliki 24), ker želimo modelirati debelino zgornje plošče mize. Izvlečemo oglišča ali angl. Extrude Vertices tako, da pritisnemo tipko **[E]**, potem pritisnemo še tipko **[Ctrl]**, da omejimo gibanje miške po kvadratnih mreže. Premaknemo miško navzgor npr. za 1 enoto (to je višina oz. debelina zgornje plošče mize, ki je lahko tudi večja - desna slika 24). Pri tem se na zaslonu izriše modra črta, z-os, po kateri lahko delamo izvlek (srednja slika 24).



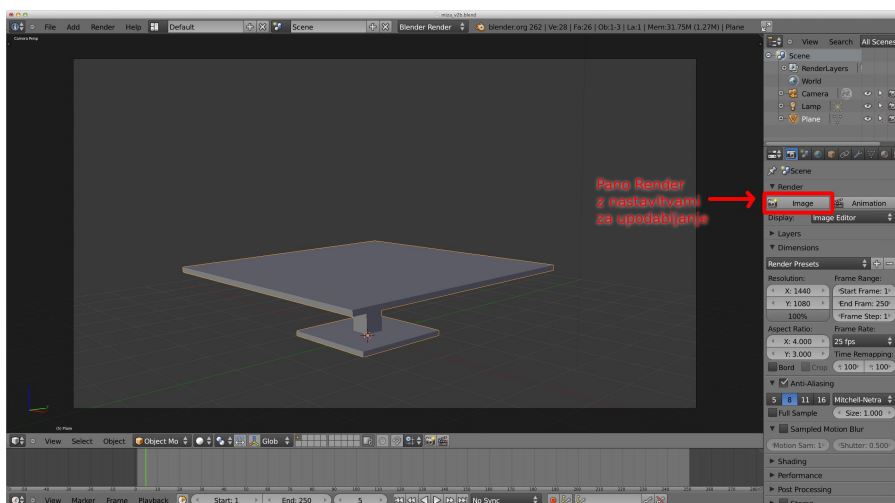
Slika 24: Pred izvlekom debeline zgornje plošče mize, med njim in po njem

V 3D-oknu se prestavimo s klikom na kolesčku miške in njenim premikom tako, da si nastalo mizo bolj podrobno ogledamo v perspektivnem pogledu (slika 25). S pritisnjenim kolesčkom miške in vlečenjem le-te se lahko gibamo tudi okrog cele mize.



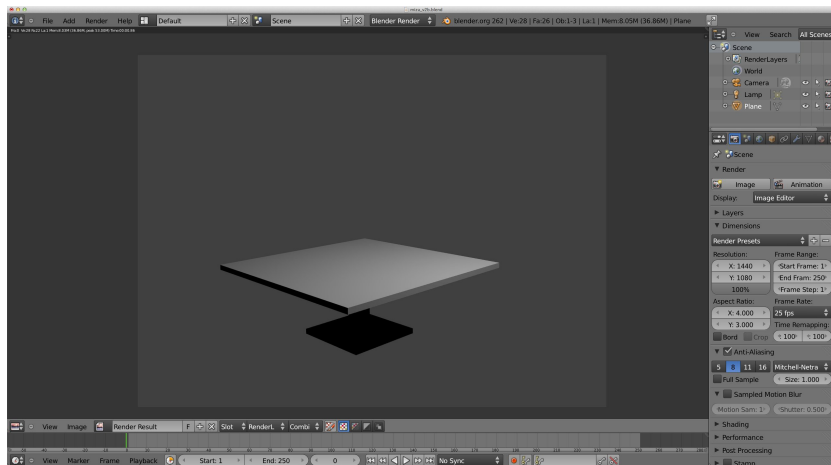
Slika 25: Perspektivni pogled na mizo in kamero v oknu z imenom 3D-pogled oz. (angl. 3D View)

Položaj kamere in njene nastavitve v sceni določata sliko, ki jo program Blender izračuna pri upodabljanju (angl. Render) ob pritisku na gumb Image (Slika), ki je v panoju Render (slika 26).



Slika 26: Pogled kamere z označenim gumbom Image, ki je v panoju Render

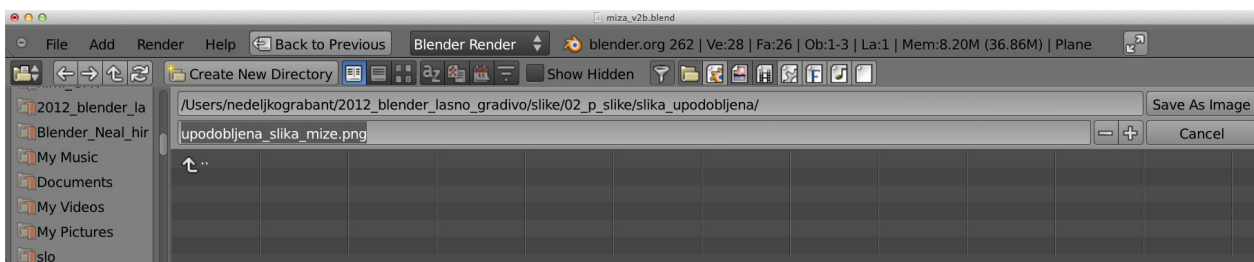
Drugi način vklopa upodabljanja oz. izračuna slike je s pritiskom na tipko **[F12]**, s čimer se okno 3D-pogled spremeni v okno UV/Image Editor (UV-urejevalnik slik) z upodobljeno sliko mize (slika 27).



Slika 27: Okno UV/Image Editor z upodobljeno sliko mize

Tako nastalo sliko shranimo v datoteko (v slikovnem formatu PNG) tako, da pritisnemo na gumb **[F3]** in v novem oknu izberemo na levi strani mapo ter v vnosno polje, ki je levo od

gumba **[]**, vpišemo ime datoteke npr. upodobljena_slika_mize.png (slika 28).

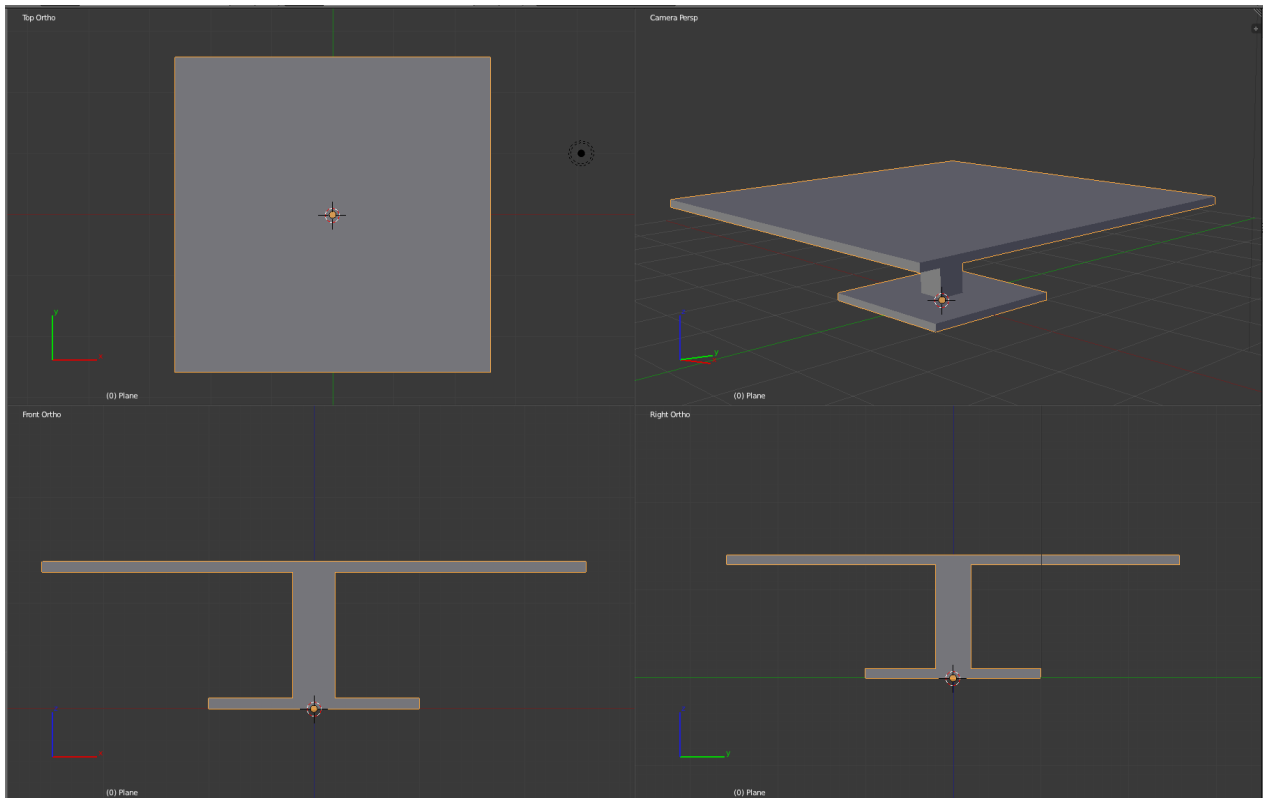


Slika 28: Pogovorno okno za shranjevanje upodobljene slike

Preklop iz okna UV/Image Editor (UV-urejevalnik slik) z upodobljeno sliko nazaj v 3D-pogled naredimo s pritiskom na tipko **[Esc]**.

Do sedaj izdelan model mize lahko shranimo z bližnjico **[Ctrl] - [S]** in pritiskom na tipko **[↵]**. Pred tem lahko povečamo številko verzije različice Blender datoteke, če ima v imenu številko (v našem primeru miza_v1.blend) s pritiskom na numerični tipkovnici na tipko **[Pad+]** (prav tako številko lahko zmanjšamo s pritiskom na tipko **[Pad-]**). Ime datoteke Blender se v našem primeru spremeni iz imena **miza_v1.blend** v ime **miza_v2.blend**.

Pogled na mizo iz treh pravokotnih smeri (tloris, naris in stranski ris) in iz pogleda kamere se preklopi s kombinacijo tipk **[Ctrl] - [Alt] - [Q]** (slika 29).



Slika 29: Štiri pogledi tloris, naris, stranski ris in pogled kamere (začetek levo zgoraj in potem v nasprotni smeri glede na urin kazalec)

Iskreno Vam čestitam za uspešno izdelan model preproste mize in za shranjevanje upodobljene slike ter modela v Blender formatu s končnico `blend`. Verjetno ste opazili, da je modeliranje relativno preprost postopek, če se poznajo osnovni postopki in logični koraki v izbranem programu za modeliranje.

Če povzamem, je v tem primeru postopek modeliranja sestavljen iz: vstavljanja ravnine (bližnjica je `⇧`-`A`), preklopa v način urejanja (`⇧`), večkratnega izvleka oglišč (`E`), preklica premikanja po izvleku (`Esc`), spremembe pogleda (numerični tipki `Pad5`, `Pad7`, `Pad1`, `Pad3`), povečave ali manjšanja pogleda (`Pad+` ali `Pad-`), vzporednega premika pogleda (`⇧` + `⇧` + premik `⇧` v zeleno smer), uporabe funkcije skaliranja (tipka `S`), upodabljanja (`F12`), shranjevanja datoteke (`F2` ali `Ctrl` - `S`) in shranjevanja upodabljanje slike (`F3`).



Vprašanja za preverjanje razumevanja poglavja 2 Hiter začetek

Pisno in/ali ustno odgovori na vprašanja v nadaljevanju.

Kaj pomeni beseda Blender in od kod prihaja?

Na spletni strani <http://www.blender.org/> poišči zanjo aktualno različico programa Blender! Kdaj

je ta bila objavljena in ali se dobijo tudi prejšnje verzije tega programa?

Napiši pot do mape, kje se v operacijskem sistemu Windows nahaja mapa s pisavami!

S katero bližnjico se preklopi med Objektnim načinom (angl. Object Mode) in Načinom urejanja (angl. Edit Mode)?

S katero bližnjico vklopi funkcija Izberi vse (angl. Select all) za vse objekte ali oglišča?



Naloge za poglobljanje poglavja 2 Hiter začetek

Pisno ali ustno odgovori na naslednja vprašanja:

Odgovore na naslednja vprašanja je možno dobiti na Blenderjevi vikipediji (<http://wiki.blender.org/index.php/Doc:2.6/Reference/>).

Preglej Blenderjo Wikipedijo in v angleščini in slovenskem prevodu naštej posamezna področja te vikipedije?

Katero področje se ti zdi najboljše predstavljeno?

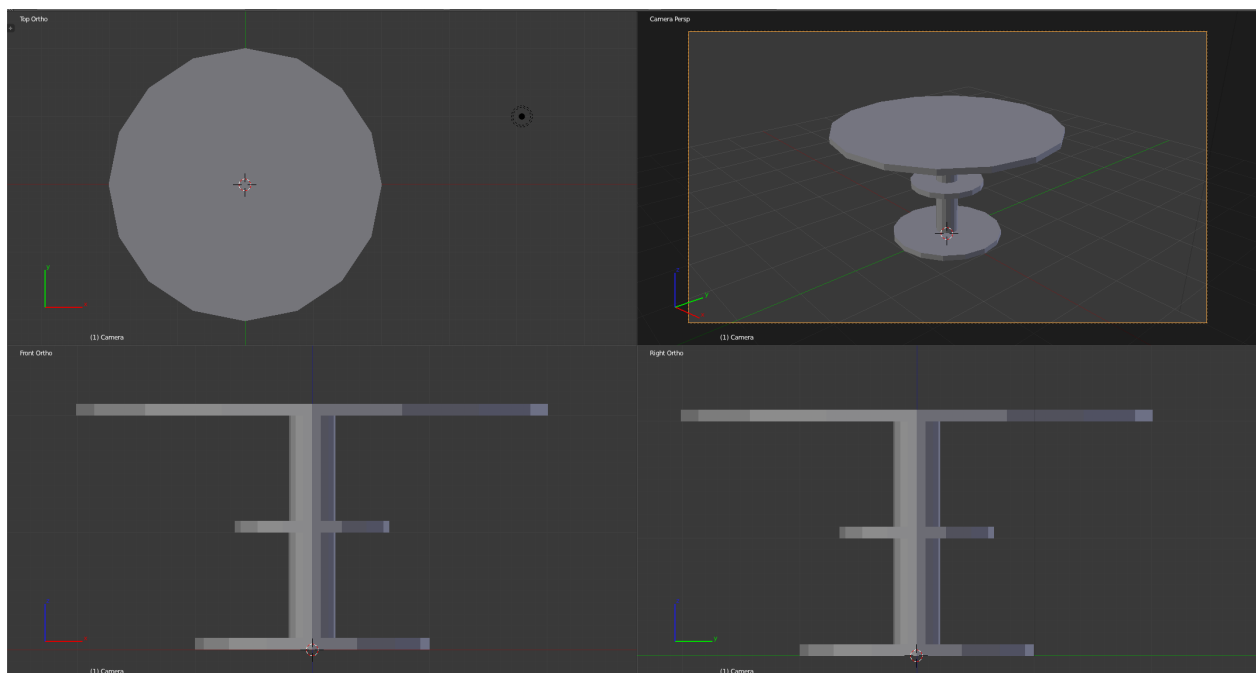
Kje se dobijo proste knjige o Blenderju?



1. Samostojna praktična naloga Blender

Iz začetnega kroga (angl. Circle) s 16 vozlišči (angl. Vertices, te nastavi v okno Object Tools (bližnjica je tipka **T** ali tipka **F6**) spodaj pod istoimenskim vnosnim poljem) ustvari z izvlekom oglišč (**E**) naslednjo mizo (slika 20).

Shrani upodobljeno sliko in datoteko Blender pod imenom datum_ime_priimek_okrogla-miza.

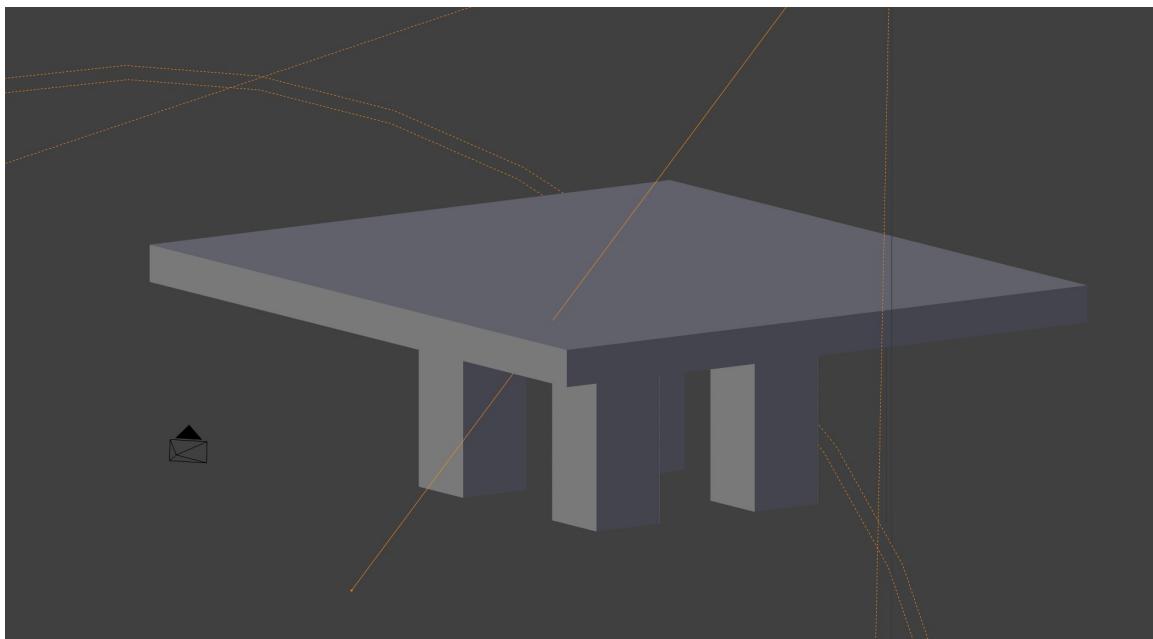


Slika 30: Pogled iz štirih načinov na šestnajstkotno "okroglo" mizo





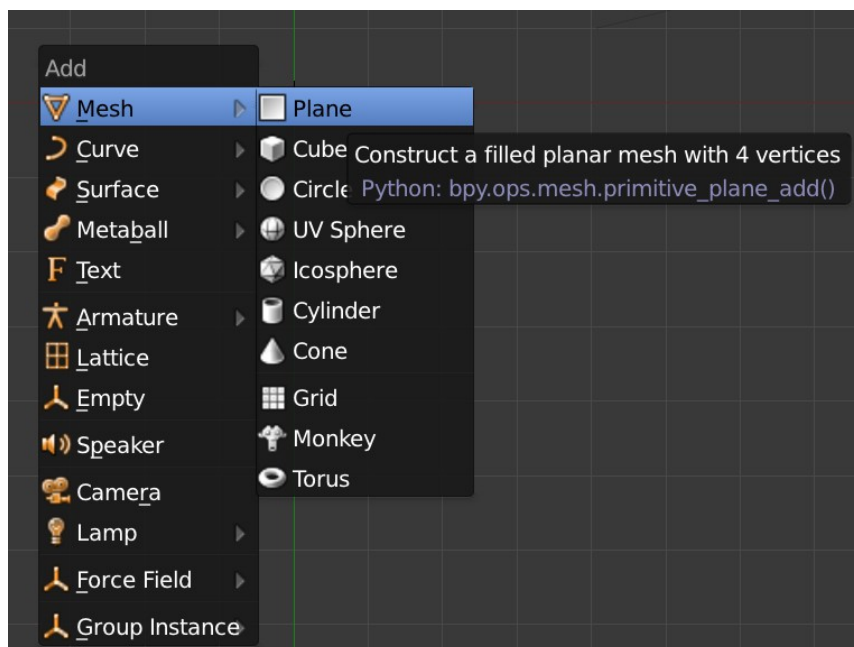
2. Samostojna praktična naloga Blender za poglobljanje poglavja 2

Namen tega vodiča je ustvarjanje pravokotne mize (slika 31). Poskusi sam, če ne gre, si lahko pomagaš z nadaljnjim opisom modeliranja. Pri modeliranju bomo postopali tako, da je model narejen iz enega dela (kosa), kar je ponavadi boljše za animacijo ali druge opravke z modelom. Drugače je potrebno posebej skrbeti za vsak posamezen del.




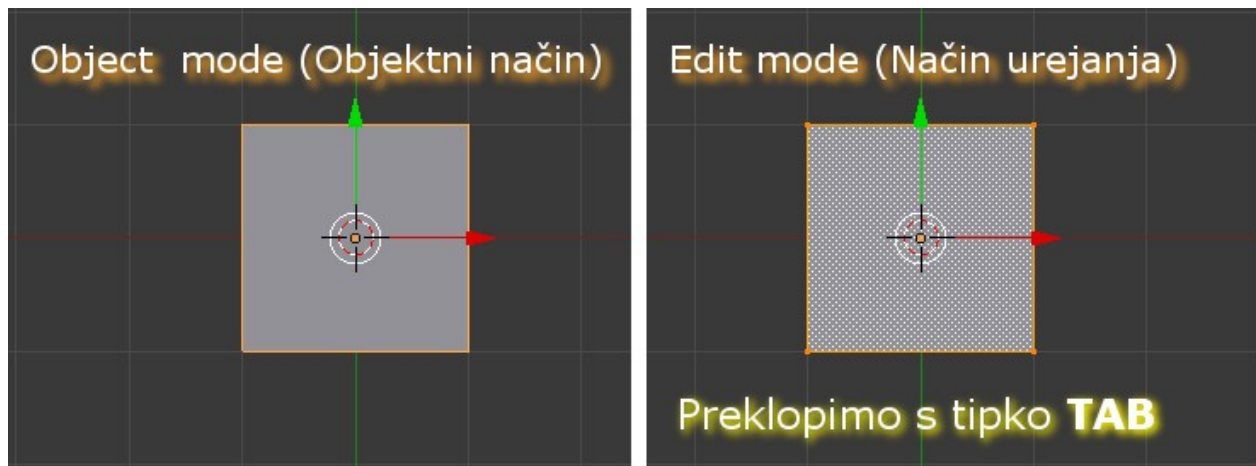
Slika 31: Primer modeliranja štirioglate mize

V začetno sceno (prizorišče) vstavi s kombinacijo tipk  +  angl. Plane (Ravnino), ki je bolj podrobna kvadratu (slika 32).



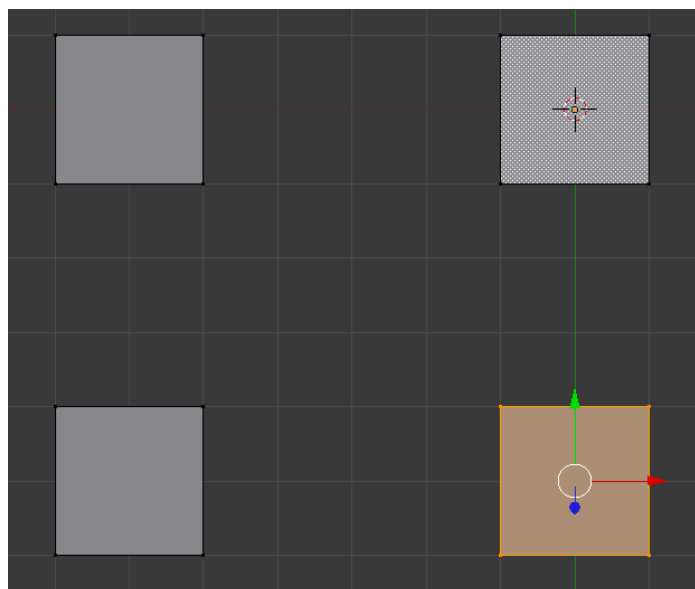
Slika 32: Vstavljanje ravnine oz. angl. Plane

S tipko TAB oz.  se prestavi iz Objektnega načina (angl. Object) mode v Način urejanje (angl. Edit mode) (slika 33).



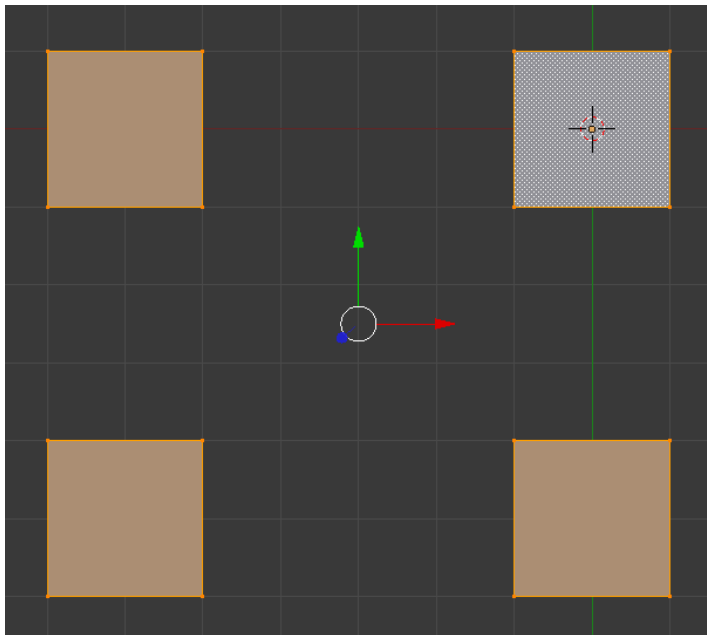
Slika 33: S tipko TAB naredimo preklomp iz objektnega v način urjenja oz. angl. Object mode v Edit mode

Označimo vsa oglišča ravnine (kvadrata) s tipko **A** in označeno trikrat zaporedoma podvojimo s kombinacijo tipk Shift oz. **⇧** + **D** ter jih ustrezno premaknemo za po štiri enote (slika 34). Za premik kopirane ravnine po mreži oz. glavnih pomožnih črtah držimo pritisnjeno tipko **Ctrl**.



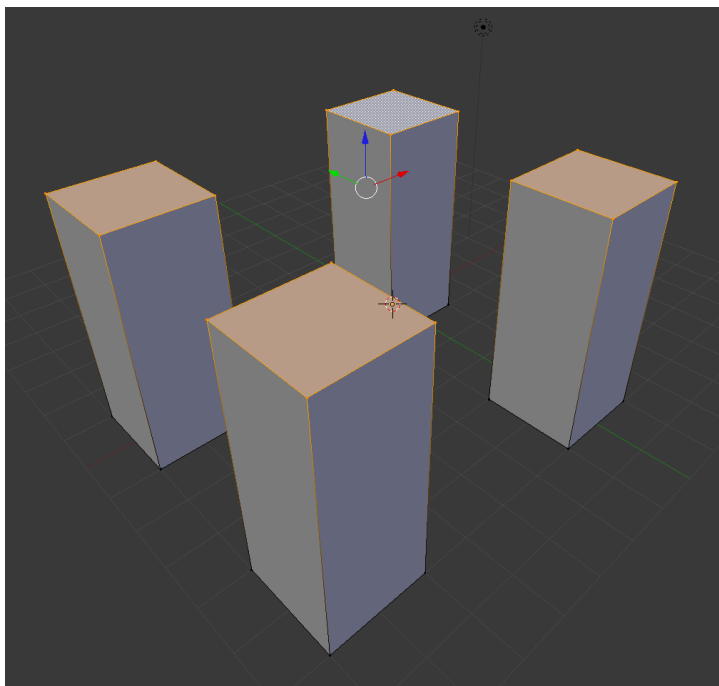
Slika 34: Po kopiranju je označena zadnja ravnina (kvadrat)

S tipko **A** izberemo vse štiri ravnine (kvadrate) (slika 35).



Slika 35: Izbrane so vse štiri ravnine (kvadrati)

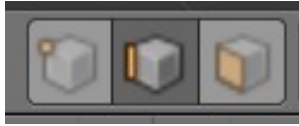
Posamezne štiri ravnine (pravzaprav so to kvadrati) bomo kot noge mize »Izvlekli« (angl. Extrude) navpično po z-osi s pritiskom na tipko **E** (slika 36) in s premikom miške navzgor. Pri tem bo najbolje, da se postavimo v vzporedno ravnino glede na premik s pomočjo tipke **Pad 1** in ob tem pogledjmo, če v zgornjem levem kotu piše »Front Ortho«. V primeru, da temu ni tako, pritisnemo še tipko **Pad 5**. Večina tipk je tu preklopnih, kar pomeni, ko enkrat pritisneš tipko (ali pogosto tudi kombinacijo tik), se vklopi funkcija in s ponovnim pritiskom tipke (ali kombinacije tipk) se ta funkcija razveljavi ali izklopi.



Slika 36: Izvlek ravnin (kvadrati) po z-osi (aplikati) za noge bodoče mize

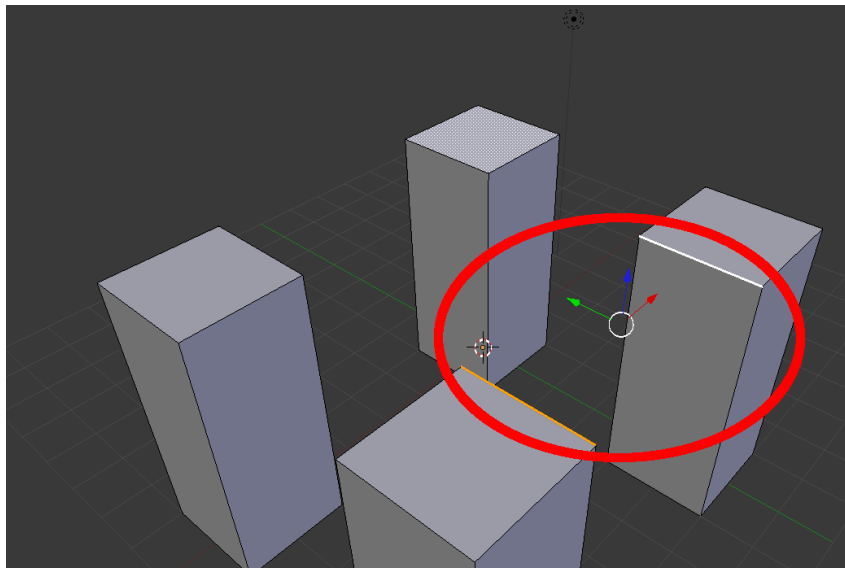
Učno gradivo je nastalo v okviru projekta Munus 2. Njegovo izdajo je omogočilo sofinanciranje Evropskega socialnega sklada Evropske unije in Ministrstva za šolstvo in šport.

S tipko **A** odstranimo izbor označenih oglišč za vse štiri ravnine in kliknemo v spodnji glavi 3D-okna na ikono Edge select (Izbor robov) (slika 37).



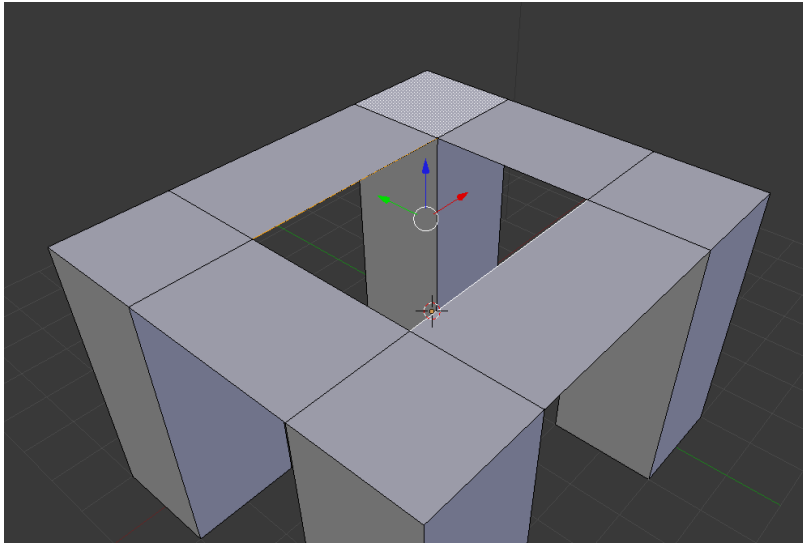
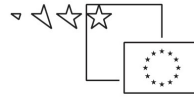
Slika 37: Izbrana je srednja ikona Izbor roba oz. Edge select

Za izbor dveh nasprotnih notranjih robov kliknemo najprej na en rob in s pridržano tipko Shift oz. **⇧** kliknemo še na drug nasprotni notranji rob druge sosednje noge (slika 38).



Slika 38: Izbrana sta dva nasprotna notranja robova dveh sosednjih nog (v rdeči elipsi)

Da ustvarimo ploskev (angl. Face) med dvema označenima roboma, je potrebno pritisniti na tipko **F**. Potem prekličemo izbor s ponovnim pritiskom na tipko A (Nič izbrano) in s klikom ter s tipko Shift oz. **⇧** izberemo naslednji nasprotni rob sosednje noge. S pritiskom na tipko **F** ponovno ustvarimo ploskev ali angl. Face. To nadaljujemo tako dolgo, da pridemo okrog modela in potem zapolnimo še dva sosednja roba sredinske pravokotne odprtine (slika 39).



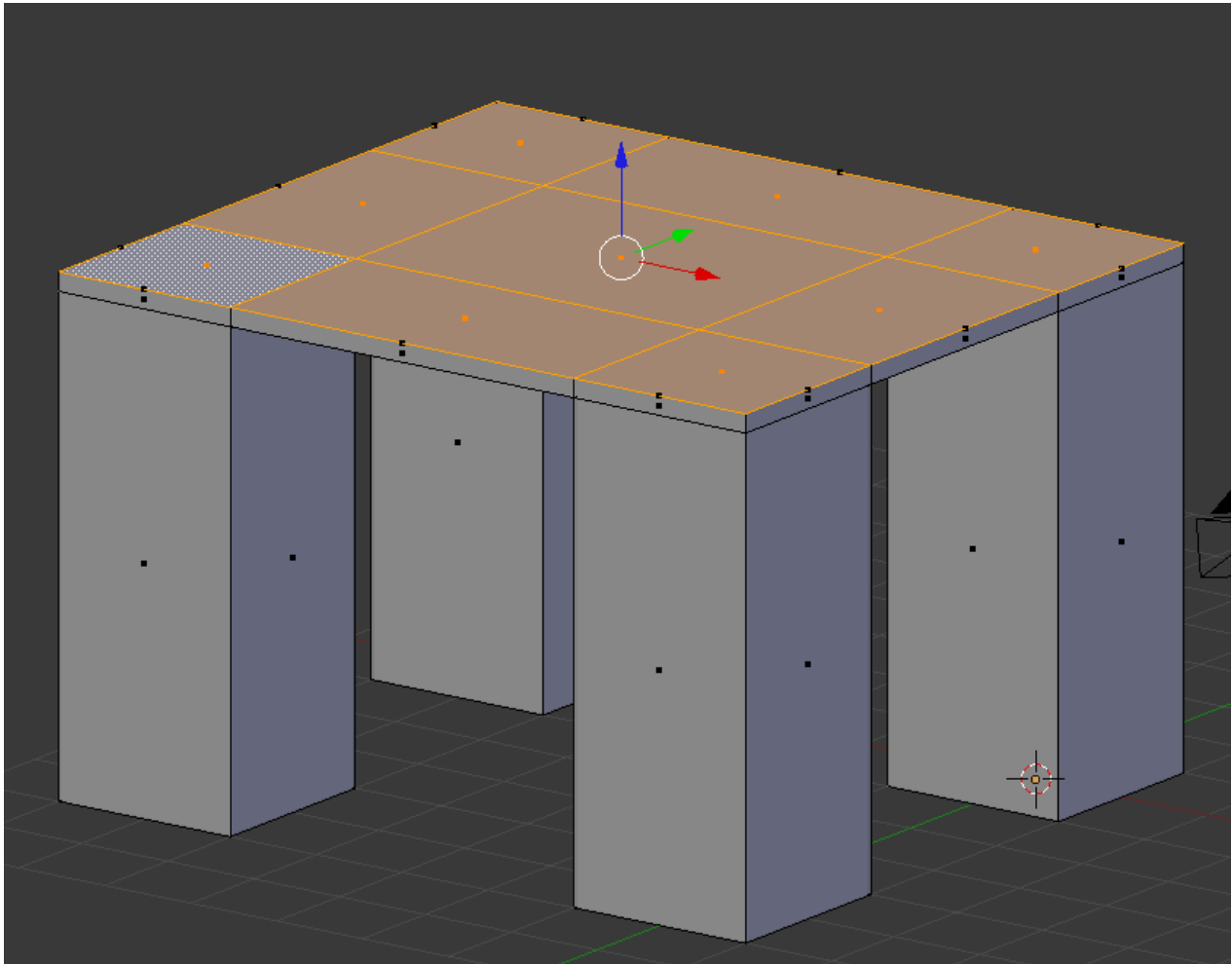
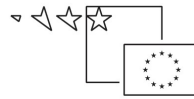
Slika 39: Skoraj "tlakovana" miza le notranja sredinska robova še nista pretvorjena v ploskev

Sedaj izberemo t. i. način Izbor ploskve s klikom na tretjo ikono Face select (slika 40).



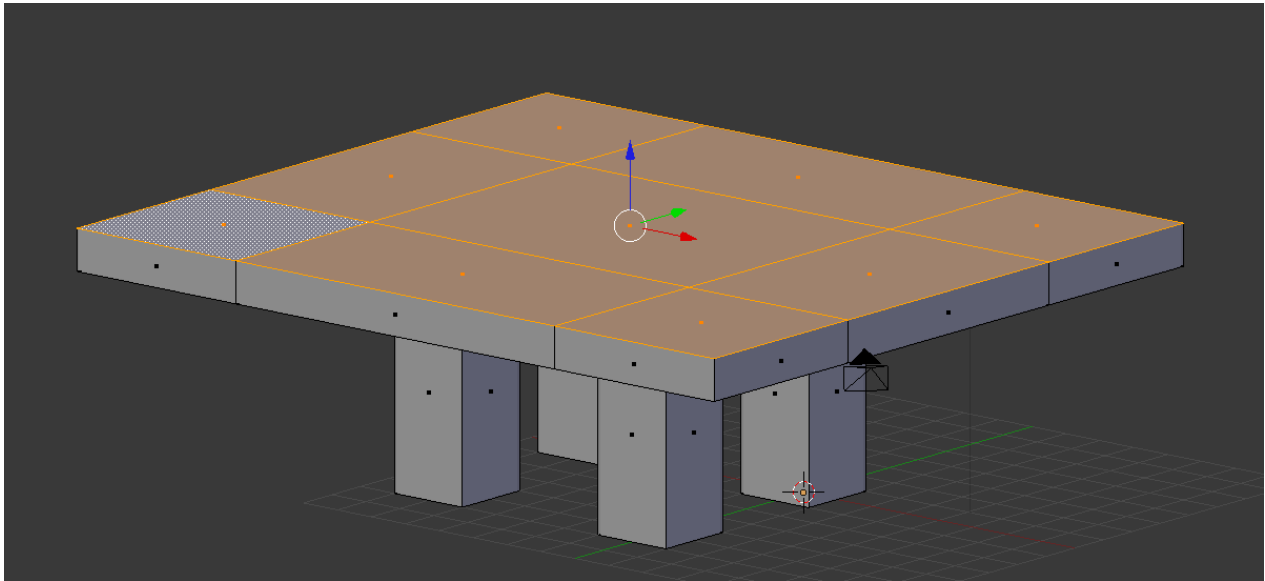
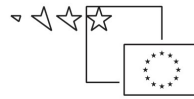
Slika 40: Izbrana zadnja ikona za izbor ploskev angl. Face select

S tipko **[C]** izberemo t. i. Krožni izbor oz. Circe select in kliknemo na posamezne ploskve, označene s sredinskim kvadratom na zgornji ploskvi. To je ravnina, ki jo želimo kot osnovni okvir izvleči in s tipko **[E]** naredimo navpičen izvlek navzgor (slika 41).



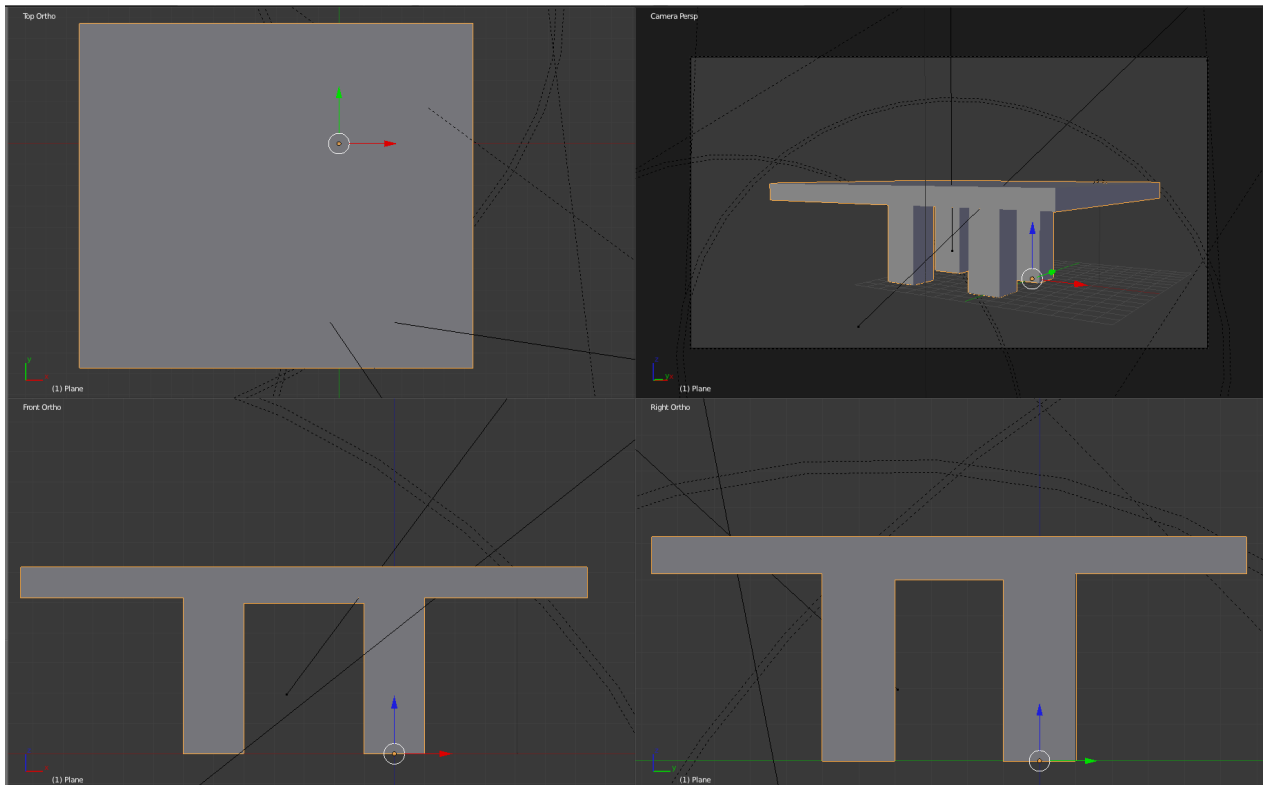
Slika 41: Osnovni okvir smo izvlekli s tipko **[E]** navpično navzgor

Zgornjo ploščo mize naredimo s ponovnim izvlekom s tipko **[E]** in s preklicem premika s pritiskom na tipko **[Esc]** ter še enim pravokotnim izvlekom s tipko **[E]** (slika 42).



Slika 42: Končan izvlek zgornje plošče mize

Na mizo lahko naenkrat pogledamo iz štirih perspektiv s pomočjo bližnjice **Ctrl** + **Alt** + **Q** (slika 43). Iz tega pogleda pridemo v prvotni pogled s ponovnim vklopom iste bližnjice **Ctrl** + **Alt** + **Q**. Shrani upodobljeno sliko in datoteko Blender pod imenom datum_ime_priimek_pravokotna-miza.



Slika 43: Pogled na mizo iz tlorisa (levo zgoraj), narisa, stranskega risa in pogleda iz kamere (desno zgoraj)

Naslednja faza pri tej mizi bi bila barvanje in teksturiranje mize, kar bomo spoznali nekoliko pozneje v tem priročniku.



Vprašanja za preverjanje razumevanja poglavja 9 Napisi in besedila

Pisno in/ali ustno odgovori na vprašanja v nadaljevanju.

Opiši, zakaj je potrebno pisati besedila v programu Blender!

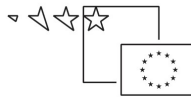
Opiši, kako je možno vstaviti in pisati besedilo v programu Blender!

Opiši pot do mape, kje v operacijskem sistemu MS Windows, MAC OS X in Linuxu najdemo pisave za besedilo programa Blender!

Opiši postopek, kako lahko v programu Blender pišemo besedilo, ki vsebuje npr. črke č,ž,š, đ, ©, @, Ä, ä, ü!

Katero funkcijo opravlja vnosno polje Extrude (Izvlak), ki je v zavihku Object Data v panelu Geometry (geometrija) - pod oznako Modification (Modifikacija), če ga spreminjamo?

Katero funkcijo opravlja vnosno polje Offset (Odmik), ki je v zavihku Object Data v panelu Geometry (geometrija) - pod oznako Modification (Modifikacija), če ga spreminjamo?



Katero funkcijo opravlja vnosno polje Depth (Globina), ki je v zavihku Object Data, v panelu Geometry (geometrija) - pod oznako Bevel (Poševno ali stožčasto), če ga spreminjamo?
Katero funkcijo opravlja vnosno polje Resolution (ločljivost), ki je v zavihku Object Data (Podatki objekta), v panelu Geometry (Geometrija) - pod oznako Bevel (Poševno ali stožčasto), če ga spreminjamo?

Opiši, kako poteka postopek barvanja objekta s poljubno barvo!

Kako se spremenimo barva izbranega objekta?

Učna situacija 2: Predstavitve uradnih projektov, izdelanih v Blenderju


S programom Blender so bili do sedaj realizirani trije filmski projekti in ena igrice (slika 44). Namen projektov je pospešiti nadaljnji razvoj tega programa in prikazati možnost, da je program primeren tudi za resne produkcijske namene. Trenutno je v drugi tretjini izdelave tudi novi peti

Blender Open Projects

Features • Feature Videos • Gallery • Movies • **Blender Open Projects** • Requirements (search)

Blender Foundation has organized four projects to validate and improve the 3D open source content creation pipeline with Blender. For each project the best artists of the Blender community were invited to come to Amsterdam to work 7-12 months in a well equipped studio on a professional and exciting target. The results and all artwork created for making it, were released under the Creative Commons.

Elephants Dream (2006)



Project Orange, the first Open Movie, started in September 2005 and premiered end of March 2006. The ambitions and targets for Orange were set very high, the character animation system had to be fully recoded, the rendering system needed a lot of upgrades, and most of all: a node-based Compositor was required.

Elephants Dream was co-produced with the Netherlands Media Art Institute, who challenged the team to explore an abstract concept and storyline. This 10 minute short shows two people who explore a strange mechanical world, with each an entire own perception on what's actually there. The emotional ending shows once more the violent impact of mixing reality and fantasy.

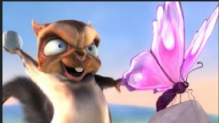
Credits

Production blog: www.elephantsdream.org

Director: Bassam Kurdali (USA, Syria)
Art Director: Andreas Goralczyk (Germany)
Lead Artists: Matt Ebb (Australia)
Lead Artists: Bastian Salmela (Finland)
Lead Artists: Lee Salvemini (Australia)
Technical Director: Toni Alatalo (Finland)

Music and sound design: Jan Morgenstern (Germany)
Producer: Ton Roosendaal (Netherlands)

Big Buck Bunny (2008)



Project Peach, the second Open Movie, started in October 2007 and premiered April 2008. Target this time was to create good hair/fur editing and rendering, more advanced support for cartoonish characters, and solve complexity in rendering outdoors environments with grass, trees and leaves.

Big Buck Bunny was the first project in the Blender Institute Amsterdam. This 10 minute movie has been made inspired by the best cartoon tradition. A giant rabbit finds his happy sunny morning walk being disturbed by three rascal rodents. In a comical grand finale he gets even with them.

Credits


Production blog: <http://www.bigbuckbunny.org>

Director: Sacha Goedegebure (Netherlands)
Art Director: Anreas Goralczyk (Germany)
Lead Artist: Enrico Valenza (Italy)
Character Rigging & Animation: Nathan Vegdahl (USA)
Character Animation: William Reynish (Denmark)

Technical Director: Campbell Barton (Australia)
Software Developer: Brecht van Lommel (Belgium)

Music and sound design: Jan Morgenstern (Germany)
Producer: Ton Roosendaal (Netherlands)

Sintel (2010)



Project Dorian was the third Open Movie, which started in september 2009 and premiered at the Netherlands Film Festival end of september 2010.

With a much larger team and bigger budget, the targets were very ambitious, resulting in a 15 minute short film that has found much acclaim on the web. Especially story telling and editing has been paid much attention to.

Development targets mainly focused on getting Blender 2.5 more stable and usable, getting sculpting tools improved, efficient detail rendering and Global Illumination rendering.


Credits

Production blog: <http://www.sintel.org>

Director: Colin Levy (USA)
Art Director: David Revoy (France)
Script: Esther Wouda (Netherlands)
Artists: Angela Guenet (Canada), Soenke Maeter (Germany), Ben Dansie (Australia), Pablo Vazquez (Argentina), Dolf Veenvliet (Netherlands), Nathan Vegdahl (USA), Lee Salvemini (Australia), Beorn Leonard (Australia/UK), Jeremy Davidson (Australia/UK)
TDs: Brecht van Lommel (Belgium), Campbell Barton (Australia)

Music and Sound design: Jan Morgenstern (Germany)
Producer: Ton Roosendaal (Netherlands)

Yo Frankie! (2008)



Project Apricot, the Open Game, started February 2008 and published the game in September 2008. This time the target was to improve Blender as a tool in the game creation pipeline, using a level editor to the external engine Crystal Space. During the project also game creation and prototyping in Blender itself was added as a target.

Yo Frankie! was also executed in the Amsterdam studio of the Blender Institute. The game was delivered in both Blender Engine as in Crystal Space. The player controls evil rodent Frankie, who explores the forest seeking for other animals to harass.

Credits

Production blog: www.yofrankie.org

Blender team:

Lead Design: Chris Plush (USA)
Lead Artist: Pablo Vazquez (Argentina)
Logic / Scripting: Campbell Barton (Australia)

Crystal Space team:

Lead Design: Dariusz Dawidowski
Logic / Level editor: Pablo Martin (Spain)
Software: Frank Richter (Germany)

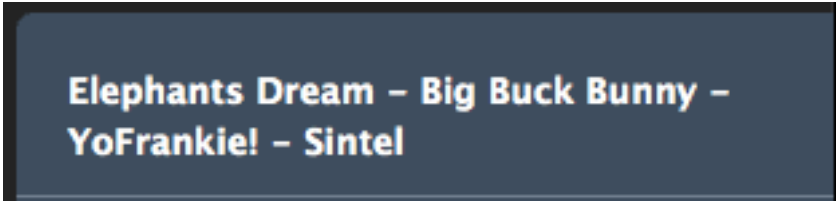
Music and sound design: Claire Fitch (Ireland)

Blender Institute support:

Blender Software: Brecht van Lommel (Belgium)
Producer: Margreet Riphagen (Netherlands)
Production: Ton Roosendaal (Netherlands)

projekt z imenom Mango. S klikom na sliko, ki je pod naslovom posameznega projekta Blender, je narejena povezava do spletišča projekta (slika 44). V nadaljevanju bodo na kratko opisani posamezni projekti.

Do teh projektov dostopate s klikom na povezavo <http://www.blender.org/features-gallery/blender-open-projects/>, 10. 6. 2012. Do te spletne strani vodi tudi slikovna povezava (slika 45), ki je dostopna z desne strani na začetni uradni spletni strani www.Blenderr.org (slika 45).



Elephants Dream – Big Buck Bunny –
YoFrankie! – Sintel

Slika 45: Povezava do spletišča z opisom uradnih projektov, izdelanih z Blenderjem

Večina vsebine na teh spletnih straneh posameznih projektov, vključno s slikami in glasbo, je prosto dostopna in jo je dovoljeno razširjati pod licenco Creative Commons (<http://creativecommons.org/>). Realni projekti, ki so jih zaključili s filmom ali igrico, so odlična osnova za učenje.

Elephants Dream

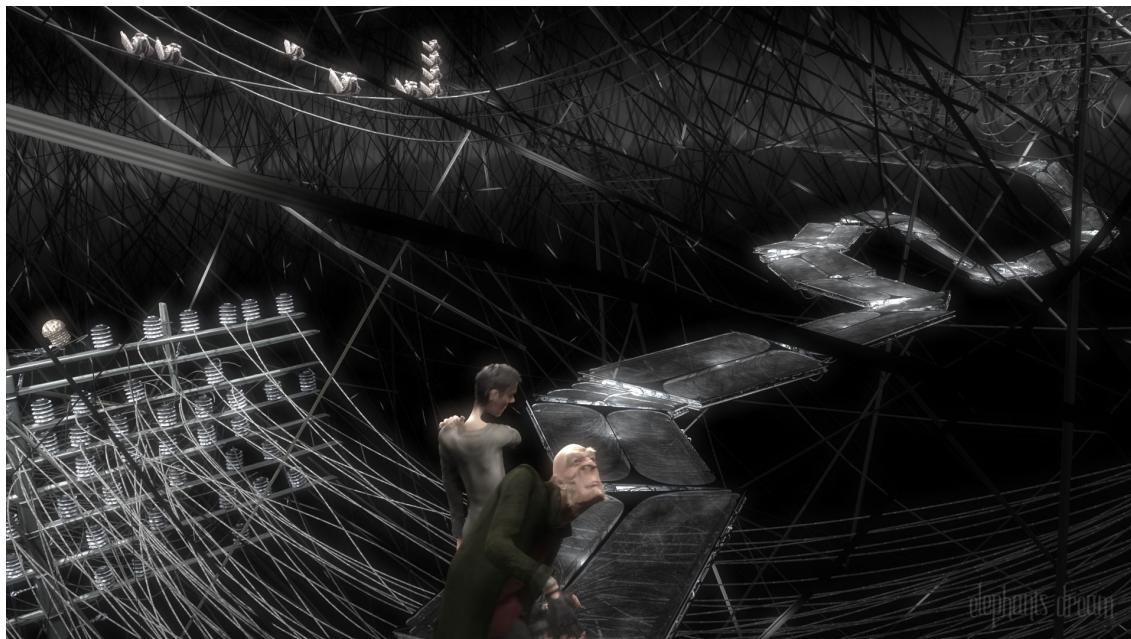
Elephants Dream s kratico ED (slika 46) je prvi Blenderjev film [3], izdelan samo iz odprtokodnih programov. Film je bil izdelan v studiu pod delovnim naslovom Orange Open Movie Project v Amsterdamu leta 2006/2007. V filmu je upodobljena zgodba o komunikaciji in fikciji, je na svetu prvi odprti 3D-animirani film.

S spletne strani www.elephantsdream.org si še vedno lahko prenesemo video in si ga ogledamo, lahko pa še vedno kupimo DVD, kjer dobimo poleg filmčka tudi produkcijske datoteke, ki so vključene vanj (modeli, scene, teksture, zvoki ...) (slika 46).



Slika 46: Spletišče projekta *Elephants Dream*

Film prikazuje dve osebi, ki raziskujeta na videz nenavaden in neskončen stroj. Proog, starejša med tema dvema, je vodič in zaščitnik stroja. Proog je razkazoval stroj in njegove nevarnosti svojemu mlajšemu prijatelju Emu, ki ga je radovedno poslušal (slika 47).



Slika 47: Proog in Emu si ogledujeta stroj

Vendar pa je bil stroj veliko več, kot je mislil Proog, zato je bilo njegovo vodenje razkazovanja stroja skrivnostno ... (slika 48).



Slika 48: Emu in Proog v dvigalu stroja (z leve na desno)

Drugače pa je Elephants Dream surrealistična zgodba med sanjami, komunikacijo in fantastiko. To pa je bil tudi prvi odprti film, za katerega smo lahko dobili modele, iz katerih bi lahko izdelali lastno igro ali kakšen drug filmček. Teh datotek je okoli 7 GB, kar je za tako kratek film veliko.



Slika 49: Ena od ključnih scen z začetkom prepira iz filma Elephants Dream [1]

V kratkem animacijskem filmu je lepo razvidno, kako se lahko v Blenderju dobri umetniki poigravajo z osvetlitvijo in dosežajo odlične učinke. Tu je bil tehnično poudarek na "post procesiranju" oz. obdelavi posnetkov po upodabljanju (slika 50). Ambicije in cilji za projekt Orange so bili postavljeni zelo visoko in z namenom ustvariti animacijo sistema osebkov (angl. character animation system), ki se lahko v celoti snema. To je dober sistem za upodabljanje s potrebno nadgradnjo takratnega sistema (predvsem z vozlišči, ki temeljijo na kompozitorju, angl. node-based Compositor).



Slika 50: Naslovna stran DVD

Več o filmu in vse slike tega dela so s spletišča www.elephantsdream.org, 10. 5. 2012 (slika 46). Oktobra 2010 je Nemeč Wolfgang Draxinger na konferenci Blender predstavil upodabljanje ED v stereo 3D-tehniki. Ta stereo oz. 3D-filmska različica je dostopna na YouTube, Vimeo ali zgornjem naslovu spletišča ED.

Big Buck Bunny

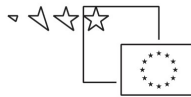
Film Big Buck Bunny (slika 51) so začeli pripravljati in delati leta 2007 [3]. Uradno je izšel 10. aprila 2008 v Amsterdamu, na spletu se je pojavil 30. maja 2008 kot drugi odprti film za filmom Elephants Dream.



KONZORCIJ ŠOLSКИH CENTROV



REPUBLIKA SLOVENIJA
MINISTRSTVO ZA ŠOLSTVO IN ŠPORT



Naložba v vašo prihodnost
OPERACIJO DELNO FINANCIRA EVROPSKA UNIJA
Evropski socialni sklad

THE PEACH OPEN MOVIE PROJECT PRESENTS



Big Buck BUNNY

WRITTEN AND DIRECTED BY SACHA GOEDEGEBOURE - ART DIRECTOR ANDREAS GORALCZYK - LEAD ARTIST ENRICO VALENZA
ANIMATORS NATHAN VEGDAHL, WILLIAM REYNISH - TECHNICAL DIRECTORS CAMPBELL BARTON, BRECHT VAN LOMMEL
MUSIC BY JAN MORGENSTERN - PRODUCED BY TON ROOSEDAAL, BLENDER FOUNDATION

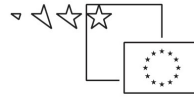
© LICENSED AS CREATIVE COMMONS 3.0 ATTRIBUTION

WWW.BIGBUCKBUNNY.ORG

SPONSORS:
creative commons nederland digital pioneers wavemage Maqina FILMMORE blenderinstitute

Slika 51: Plakat Big Buck Bunny, vir: <http://www.bigbuckbunny.org>

Zgodba filma Big Buck Bunny se odvija v gozdu, kjer zajec Buck sreča tri nasilna bitja, Franka (veverica), Rinky (lisica) in Gamera (krt). Veverica se imenuje Frankie. V gozdu se zabava tako, da s svojimi prijatelji nagaja nedolžnim živalim gozda. Ko Frank ubije dva metulja in zajca Bucka napade z lešniki in s kostanjevimi ježicami, se slednji odloči, da se bo maščeval ter vzel pravico v svoje roke ...



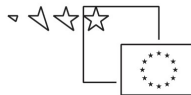
Slika 52: Trije "zlobni" glodavci Gamera, Frank in Rinky

Big Buck Bunny je pripomogel k izboljšavi programa Blender na področju las (delcev), osvetlitve in UV-mapiranja, napredna je podpora za risanje osebkov oz. karakterjev, reševanje kompleksne interakcije v zunanjih okoljih s travo, drevesi in listi.

Filmsko dogajanje je postavljeno na travnik in v gozd. V filmu so se posebej ukvarjali z napredkom sistema delcev (ang. Particles), s katerimi je ustvarjena realistična animirana trava ter dlaka (slika 52 in 53).

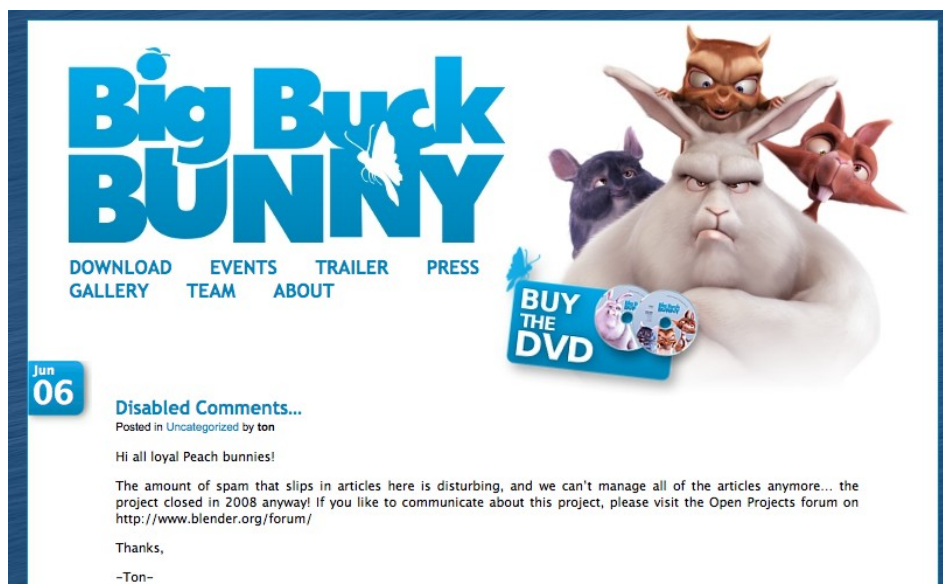
Na nizozemskem festivalu animiranega filma je Big Buck Bunny dobitnik dveh nagrad:

- Nagrada občinstva MovieSquad (otroci od 8-12 so glasovali v šolskem programu HAFF).
- Nagrada občinstva HAFF (obiskovalci so glasovali z več kot 40 nominacijami).



Slika 53: Idiličen začetek, ko Big Buck Bunny »zamaknjeno« občuduje metulja

Več o filmu in vse slike iz tega dela gradiva so s spletišča <http://www.bigbuckbunny.org>, 10. 5. 2012. Na tem spletnem naslovu se tudi brezplačno prenese film, produkcijske in druge datoteke, ki so nastale ob izdelavi tega animiranega filma (slika 54).



Slika 54: Spletišče filmskega projekta Big Buck Bunny

Računalniška igra Yo Frankie!

Avgusta leta 2008 je izšla igrlica na osnovi „glavnih igralcev“ iz filma Big Buck Bunny in se po hudobnežu imenuje Yo Frankie.

Projekt marelica (angl. Project Apricot), prva odprta igra, izdelana z Blenderjem, se je začel februarja 2008 in objavljen kot računalniška igra septembra 2008. Tokratni cilj projekta je bil izboljšati orodja in cevovod pogona za pripravo iger z uporabo zunanjega urejevalnika igralnega pogona t. i. Crystal Space (<http://www.crystalspace3d.org/>). V času trajanja tega projekta se je vmes pojavil kot dodaten cilj, da se ustvari še isti prototip igre Yo Frankie v Blenderjevem lastnem igralnem pogonu (angl. Blender Game Engine -BGE).



Slika 55: Šest glavnih karakterjev iz igre Yo Frankie iz BGE

Yo Frankie! se je razvijal v amsterdamskem studiu inštituta Blender. Igra je bila izdana najprej za igralni pogon Crystal Space in pozneje v lastnem Blenderjevem igralnem pogonu (angl. Game 36

Učno gradivo je nastalo v okviru projekta Munus 2. Njegovo izdajo je omogočilo sofinanciranje Evropskega socialnega sklada Evropske unije in Ministrstva za šolstvo in šport.

engine – slika 56).



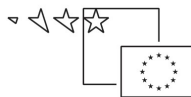
Slika 56: Scena, ko glavni junak Yo Frankie hodi čez most, posnetek je iz BGE

Igralec nadzoruje zlobnega glodavca Frankie-ja, ki raziskuje gozd in išče druge živali, da jih lahko nadleguje (slika 57).



Slika 57: Prehod Yo Frankie-ja skozi lavo, posnetek je iz BGE

Več o filmu in vse slike tega dela so s spletišča <http://www.yofrankie.org>, 10. 5. 2012 (slika 58). Na tem spletnem naslovu se lahko brezplačno prenesejo igra, produkcijske in druge datoteke, ki so nastale ob razvoju igre.



Slika 58: Spletišče projekta igre Yo Frankie

Druge igrice, ustvarjene z Blenderjem

Zahvaljujoč Blenderjevemu odprtemu konceptu se po vsem svetu posamezniki in skupine odločajo za ustvarjanje z Blenderjem v različnih projektih, med temi so tudi računalniške igrice.

38

Učno gradivo je nastalo v okviru projekta Munus 2. Njegovo izdajo je omogočilo sofinanciranje Evropskega socialnega sklada Evropske unije in Ministrstva za šolstvo in šport.

Sledijo sezname s povezavo do kakovostnih igrice, izdelanih v Blenderjevem igralnem pogonu t. i. BGE, ki jih je zbral razvijalec igrice, ki se predstavlja pod psevdonimom Ross.

Seznam končanih igrice, ki jih je izdelal Ross in z njimi služi denar:

[Color Cube](#)

[Dead Cyborg](#) (slika 59)

[Nicoles Nel's collection](#)



Slika 59: Primer spletišča za igrico Ded Cyborg

Seznam končanih igrice Blender:

[Yo Frankie](#)

[Gtown](#) (delno končana)

[Super Blender Galaxy](#)

[Mini Golf](#)

[Nanoshoooter](#)

[Tank Wars](#)

[Z-Virus 1](#)

39

Učno gradivo je nastalo v okviru projekta Munus 2. Njegovo izdajo je omogočilo sofinanciranje Evropskega socialnega sklada Evropske unije in Ministrstva za šolstvo in šport.

[Hummer Vs Heli](#)

[Grid](#)

[More than Duty](#)

[Faith](#)

Boro Toro je ena od najboljši BGE-igric: <http://www.youtube.com/watch?v=-swZUSILt8M&feature=plcp>

Seznam številnih znamenitih igric, ki jih je Ross začel izdelovati:

[1.7.1 Código Criminal](#)

[Novus Terra](#)

[Project Cassandra](#)

[BUERAKI](#)

[Krum](#)

[Soldier Of](#)

[Sparky](#)

[A Peach Story](#)

[Matter](#)

[Bongo Hop](#)

[Z-Virus 2](#)

[Quantum](#)

[Necrosys](#)

[Charlies Big Adventure](#)

[Dead Cyborg](#)

[Robot Game](#) [prototip v BGE]

[Pick A Pix 3D](#)

[Third Person Game](#) (zelo simpatična)

[Dino Crisis 4](#)

[Orbito](#)

[Faith II](#)

Seznam še več igric je dostopen na Rossovem spletišču - <http://bgame.anicator.com/> in še na - <http://www.tutorialsforblender3d.com/Demos/Demos.html> ter <http://etyekfilm.hu/makingofignite/>

Igrici "Gasoline-Racing" in "Billard-Français" sta dostopni na spletnem naslovu: <http://terrier.infographie.free.fr/index.php?id=451>

Sintel

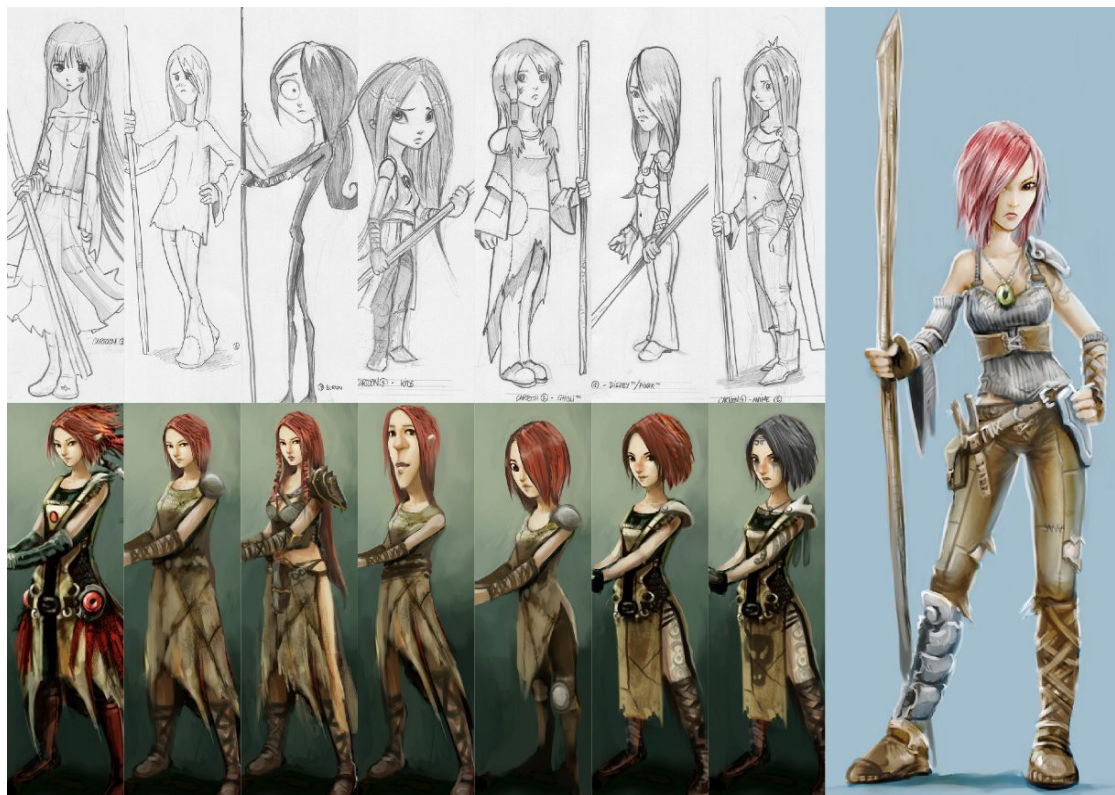
Zadnji kratki odprti računalniški animirani film, nastal v inštitutu Blender (Blender Foundation), se imenuje Sintel¹ (slika 60). Z izdelavo te animacije so pričeli maja leta 2009, uradno pa je bil izdan 27. septembra 2010 na nizozemskem filmskem festivalu.



Slika 60: Poster animiranega filma "Sintel", vir: www.sintel.org

Zgodba tega animiranega filma se dogaja okrog dekleta z imenom Sintel (slika 61), ki išče malega zmaja, ki ga je poimenovala Scales. Sintel je zmaja našla poškodovanega na ulici. Skrbela je zanj, ko se je naučil leteti, ga je ugrabil drug zmaj. Na koncu Sintel najde velikega zmaja z enim mladičem. Z njim se spopade in ga ubije...

¹ <http://www.sintel.org>



Slika 61: Koncept razvoja glavne junakinje Sintel, avtor David Revoy, vir: www.sintel.org

Ta kratki animirani film je bil narejen le s pomočjo odprtokodnih programov, primarno z Blenderjem, ki ga je razvila omenjena fundacija. Skozi razvoj tega filma se je razvijala tudi različica Blenderja 2.5 (slika 62), ki se je ob razvoju tega filma že skoraj dokončala in je bila v zadnjih fazi razvoja pred izidom stabilne različice 2.6.

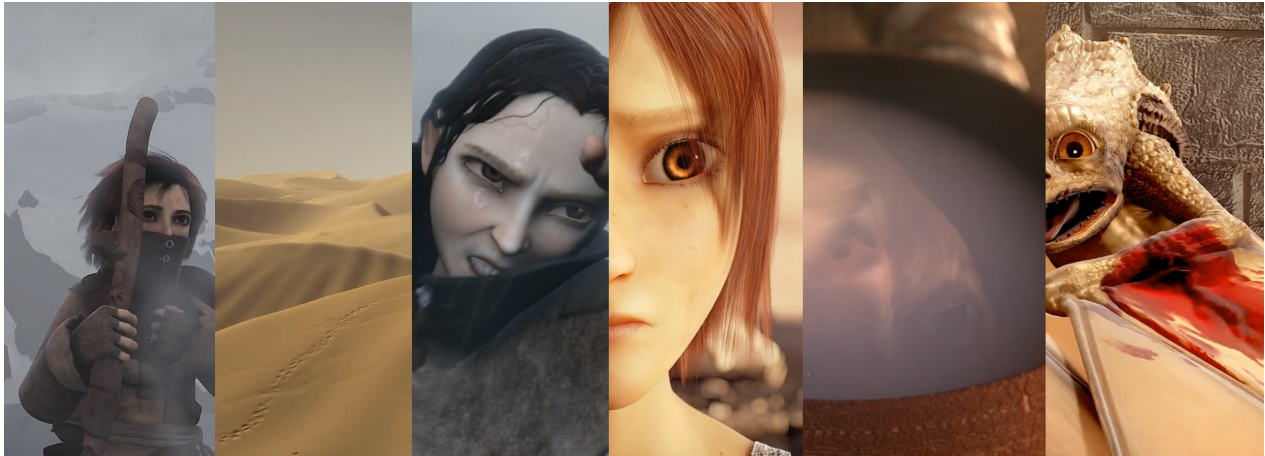


Slika 62: Pozdravno ali angl. Splash grafika iz filma Sintel za različico Blender 2.53

Za izdelavo te animacije so porabili skupno okoli 400.000 €, animacija je dolga 14 minut in 48 sekund, izdana je bila v angleškem jeziku, njen producent je bil Ton Roosendaal, njen direktor pa

Colin Levy.

V filmu sta lepo prikazana možnost ter razvoj funkcij ustvarjanja snega, sneženja, dežja, krvi, odsevov ter deževanja, fizika las in drugih sodobnih posebnih filmskih učinkov (slika 63).



Slika 63: Primer uporabe posebnih učinkov v Blenderju

Več o filmu in vse slike tega dela so s spletišča <http://www.sintel.org>, 10. 5. 2012 (slika 64). Na tem spletnem naslovu se brezplačno dobi istoimenska igra, produkcijske in druge datoteke, nastale ob razvoju projekta.



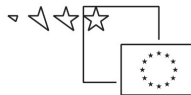
Slika 64: Spletišče o filmskem projektu Sintel, vir: www.sintel.org

Mednarodne nagrade za film Sintel

Časopis 3D Worl je v septembru 2011 (<http://www.3dworldmag.com/>) izbral film Sintel za najboljši kratki animirani film leta 2011 (slika 65).

43

Učno gradivo je nastalo v okviru projekta Munus 2. Njegovo izdajo je omogočilo sofinanciranje Evropskega socialnega sklada Evropske unije in Ministrstva za šolstvo in šport.

**Creative Awards**

CG Animation of the Year (Long Form)

CG Animation of the Year (Short Form)

Winner: Sintel (The Blender Foundation, dir. Colin Levy)

<http://www.sintel.org>

"Sintel" Trailer, Durian Open Movie Project



It's not just the Hollywood elite who are up for an award: this category is in appreciation of indie, hobbyist and student shorts, which showcase their technical prowess, animation technique, direction and editing skills.



“ Our open movies are always exciting experiments, trying to push Blender beyond boundaries. For Sintel, the team was extra-challenged by having to use the alpha version of Blender 2.5 from scratch.

Technically, the film had to accept some glitches, but in its creative execution - especially direction and storytelling - it's outstanding. Thanks for the CG short film award!

Ton Roosendaal, chairman, Blender Foundation

Contacts

Steve Jarratt
Editor, 3D World Magazine
Steve.Jarratt@futurenet.com



Andy Price
Online Editor, 3dworldmag.com
Andy.Price@futurenet.com



Ross Arthurs
Account Sales Manager
Ross.Arthurs@futurenet.com

Twitter feed

Check out MacUpdate's new Mac app bundle and get MacFonts free! (\$39.99 fonts collection - no purchase necessary!) <http://t.co/VmijvN5>
5 hours ago · reply · retweet · favorite



Join the conversation

Slika 65: V letu 2011 je časopis 3D Worl izbral film Sintel za najboljši kratki animirani film leta 2011

Na prestižnem mednarodnem filmskem festivalu je glasba iz filma Sintel in pesem "I move On" (ki jo prevedemo kot Korak naprej, slednja pesem se predvaja na koncu filma Sintel ob navajanju sponzorjev) osvojila nagrado Jerry Goldsmith (<http://www.jerrygoldsmithonline.com/>, http://en.wikipedia.org/wiki/Jerry_Goldsmith). To pomeni veliko prelomnico in nagrado za nemškega skladatelja Jana Morgensterna in pisca besedila Estherja Wouda.

Za najboljšo prenovno uporabniškega vmesnika in 3D-programa za leto 2011 je bil med 14 programi s strani časopisa 3D Worl izbran Blender (slika 66).



3D WORLD CG AWARDS 2011
THE BEST OF THE CG INDUSTRY

HOME JUDGES SPONSORS

About the 3D World CG Awards 2011
Join the celebration of the technology and artistry behind the global CG industry

The world's leading magazine for 3D artists and animators, 3D World has been at the heart of the CG industry for over 10 years. Now we are proud to lead a celebration of the artistry and technology that continues to wow audiences the world over. The 3D World CG Awards will recognise the work of the entire CG entertainment industry: from software and hardware developers who give us the tools and engines to drive creativity to the artists and technicians involved in pushing the boundaries of CG illustration, animation, visual effects and arch viz.

Winners announced

Thanks to everyone who voted in the inaugural 3D World CG Awards 2011. We announced all 14 winning entries on 14th September 2011 and you can see them below. Don't forget to buy your special awards edition of 3D World, on sale at all good newsagents and supermarkets on 14th September - or download the digital edition via www.zinio.com/3dworld-single

Technology Awards

New Application of the Year
Software Update of the Year

Winner: Blender 2.5 (The Blender Foundation)
<http://www.blender.org>

Blender 2.5

Newsletter

Become part of the 3D World community today and have privileged access to the latest news, offers, competitions and inspiration from the best-selling international magazine for 3D artists.

Get the newsletter

Subscribe

3D World is the best-selling international magazine for CG artists, covering the fields of animation, VFX, games, illustration and architecture.

Subscribe now

Contacts

Steve Jarratt
Editor, 3D World Magazine
Steve.Jarratt@futurenet.com

Andy Price
Online Editor, 3dworldmag.com
Andy.Price@futurenet.com

Ross Arthurs
Account Sales Manager
Ross.Arthurs@futurenet.com

Twitter feed

Slika 66: S strani časopisa 3D World je bil Blender izbran za najboljše prenovljeni 3D-program za leto 2011

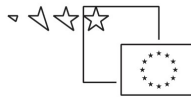
Mango

Trenutno je v pripravi oz. končni fazi najnovejši filmski projekt z imenom Mango (slika 67).

Oct 2011	Nov 2011	Dec 2011	Jan 2012	Feb/Mar 2012	Mar/Apr 2012	Aug/Sep 2012
Concept/script development	Call for portfolio submissions	Artists selected for Mango Start community fund-raiser by pre-sale of Mango DVD box set	Coder sprint/marathon to implement mango-features	Artists team starts Full animatic and previz	Filming	Premiere and online launch DVD authoring and printing

Slika 67: Časovni načrt izvedbe projekta Mango

Pri snemanju uporabljajo netradicionalni način izvajanja filmske produkcije, ker se realni posnetki mešajo z objekti, nastalimi s 3D-modeliranjem in 3D-animacijo (slika 68).



Slika 68: Robot Trimbal

Pri projektu gre za kratek znanstveno fantastičen film, ki se dogaja v Amsterdamu in ima t. i. posebne vizualne učinke ali VFX-učinke (angl. **Visual effects** (pogosta kratica je angl. **Visual F/X** ali **VFX**²). Projekt Mango temelji na filmu z resničnimi akterji in akcijami ter traja 3-5 minut.

Za projekt so bili izbrani naslednji tehnični cilji:

- Sledenje kamere in sledenje gibanja (angl. Camera and motion tracking).
- Foto-realistično upodabljanje (angl. Photo-realistic rendering).
- Izboljšan barvni cevovod (angl. Improved color pipeline).
- Izboljšana komponiranje, urejanje (video/VFX), maskiranje in razvrščanje.
- Ogenj/dim/volumen in eksplozije (angl. Fire/smoke/volumetrics & explosions).
- Urejanje Blenderjevega odvisnega grafa (angl. Blender deps-graph).



Slika 69: Umetniški izdelek glavnega grafika Davida Revoya

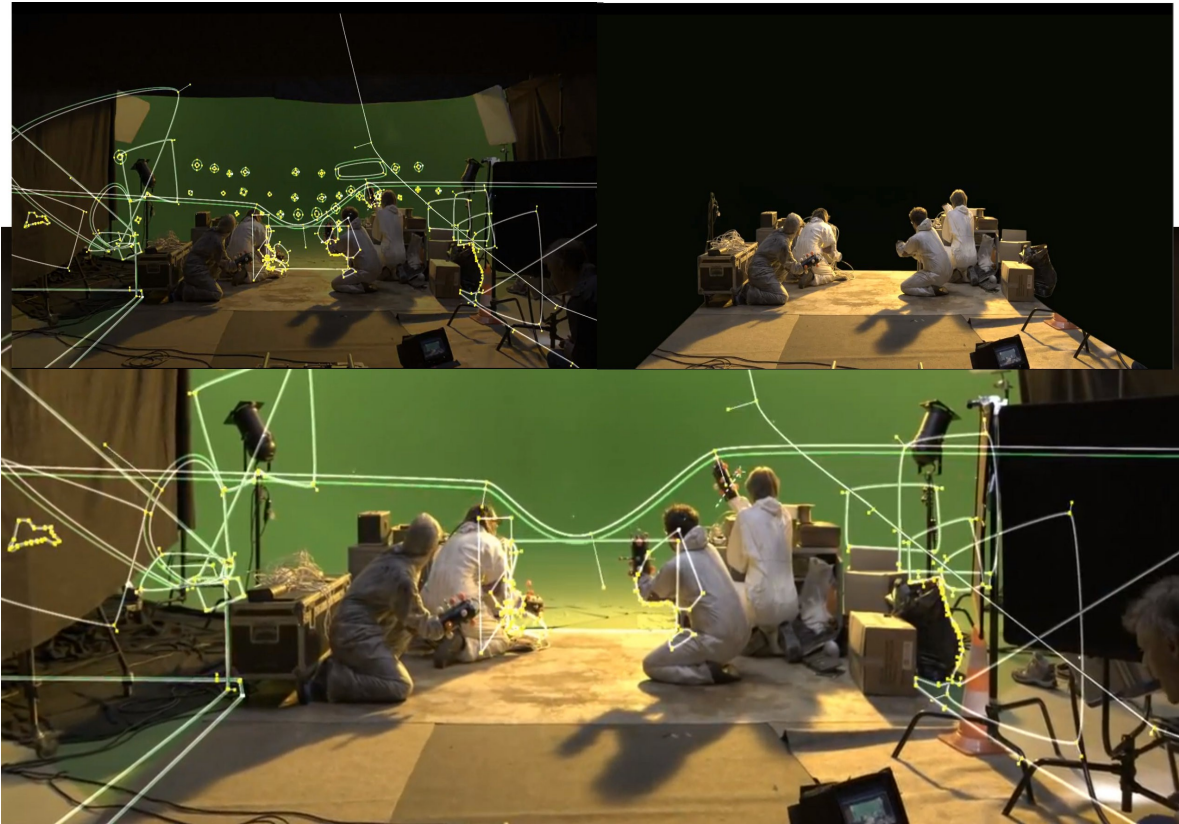
² Več o vizualnih učinkih je na spletni strani http://en.wikipedia.org/wiki/Visual_effects

Pri dosedanji klasični produkciji so bili veliki stroški pri izdelavi scen in najemu ogromnih filmskih studijev, kar je možno zmanjšati s kombinacijo virtualnih in realnih akterjev, scen in likov (slika 70).



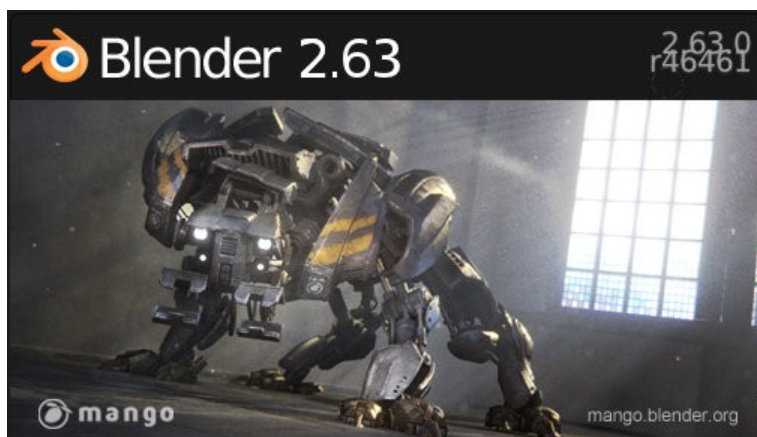
Slika 70: Sestavljen virtualni in realni svet za sceno streljanja med roboti in ljudi v filmu Mango

Pri prepletanju realnih in virtualnih svetov se uporablja zeleno platno, maskiranje ... (slika 71).

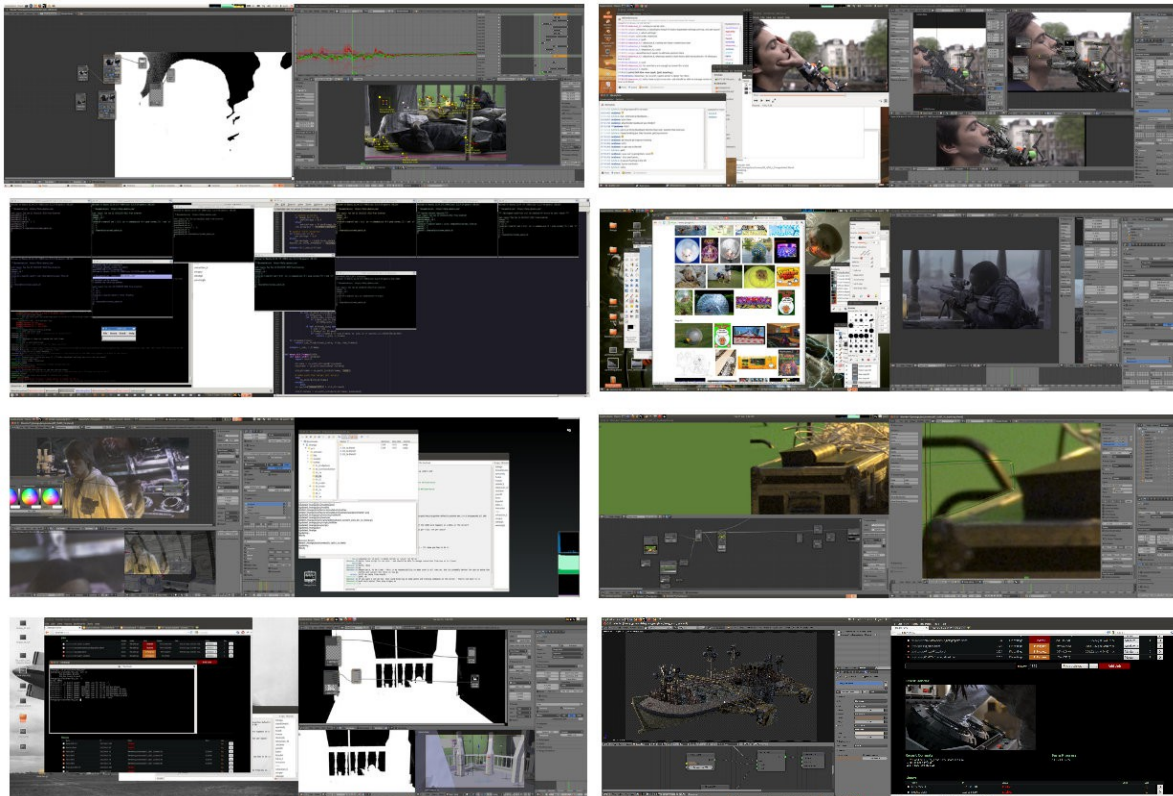


Slika 71: Uporaba rotoskopinga, masiranja in ključnih oznak v filmu Mango

V scenah se pojavljajo roboti, ki ogrožajo človeštvo. Ogroženost človeka s strani stroja je klasična tema (slika 68, 70 in 72).

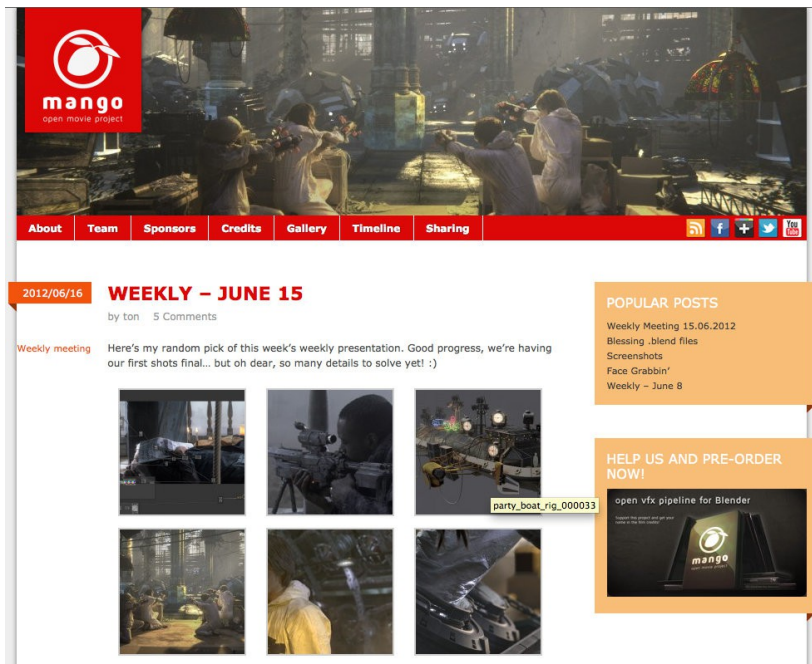


Slika 72: Logotip za Mango je viden v pojavni (angl. splash) sliki ob zagonu Blenderja
Nekaj slik iz same produkcije je vidnih v nadaljevanju (slika 73).



Slika 73: Nekaj zaslonskih slik iz delovnega procesa v projektu Mango

Več o filmu in vse slike tega dela so s spletišča <http://mango.blender.org>, 16. 5. 2012 (slika 74). Na tem spletnem naslovu bo mogoče brezplačno dobiti film, produkcijske in druge datoteke, nastale ob projektu.



Slika 74: Spletna stran projekta Mango, vir: <http://mango.blender.org>

Na vsakih nekaj dni je na spletu video ali slikovna novica s kratkim opisom napredka filma Mango (slika 75).



Slika 75: Novica o zaključnih delih pri delu v kompozitorju

Eden od glavnih ciljev projekta je izboljšava in razvoj postobdelav scene v kompozitorju (slika 76).



Slika 76: Postobdelava scene v kompozitorju

Možnosti postobdelave in delo v kompozitorju omogočajo izdelavo posebnih učinkov, ki se lepo prikazujejo na video posnetku (slika 75, 76 in 77).



Slika 77: Med nastajanjem filma se sprti prikazujejo postopki izdelave



Vprašanja za preverjanje razumevanja:

Najtej, katerih 5 uradnih projektov je izdelanih v Blenderju?

Opiši projekt Elephants Dream in razloži, kaj je bil namen tega projekta glede tehničnega razvoja programa Blender?

Opiši projekt Big Buck Bunny in razloži, kaj je bil namen tega projekta glede tehničnega razvoja programa Blender?

Opiši projekt Yo Frankie in razloži, kaj je bil namen tega projekta glede tehničnega razvoja programa Blender?

Opiši projekt Sintel in razloži, kaj je bil namen tega projekta glede tehničnega razvoja programa Blender?

Opiši projekt Mango in razloži, kaj je bil namen tega projekta glede tehničnega razvoja programa Blender?

Katere nagrade je dobil program Blender oz. kateri izdelki so bili narejeni z Blenderjem?



Naloge za poglobljanje:

Doma si oglej film Elephants Dream in odgovori na naslednja vprašanja:

Kateri so glavni junaki in kaj je sporočilo filma?

Ali se ti postavljajo kakšna vprašanja glede filma?

Kako bi se lahko film nadaljeval? Napiši.

Pred objavo filma so razmišljali o naslovu V stroju - angl. »In the machine« - zakaj so tako razmišljali?

Ali imaš željo in idejo, da bi sam sodeloval ali izdelal takšen filmski projekt?

Kateri podoben film (naslov) s podobnim sporočilom si nazadnje gledal?

Na enak način ponovi zgornja vprašanja za preostale filme in igrice, realizirano z Blenderjem. Za posamezen film, narejen z Blenderjem, izberi ustrezno glavno zvrst in če je smiselno tudi podzvrst filma:

- Pripovedna oblika: komedija, filmska melodrama, tragikomedija, tragedija.
- Razpoloženje: akcijski film, film noir (črnogledi), ljubezenski film, romanca, triler, grozljivka, erotični/porno film.
- Akcija: pustolovski film, prikaz katastrofe, kriminalka.
- Zgodovinski čas, prostorsko in družbeno pogojeni: fantazijski film, film skrivnost, družbene študije, znanstveno-fantastični film/utopični film, domači film, kavbojski film, zgodovinski film, vojni film/protivojni film, film z borilnimi veščinami/vzhodni film.

Za posamezen film, narejen z Blenderjem, ustrezno izberi druge vidike filmskih žanrov, ki so združene po slogu in/ali glede na trend:

Tehnične značilnosti

- Animacija/risanke
- 3D-film
- Nemi/zvočni film
- Črno-beli film/barvni film

Slogi in trendi

- Ekspresionistični film
- Italijanski novo realistični film
- Novi val (fr. Nouvelle Vague)
- Novi holivudski film
- Pesniško-realistični film

Pogoji produkcije:

- Avtorski film
- Neodvisni film
- Monumentalni film

Ciljna skupina

- Otroška film
- Mladinski film
- Ženski film
- Družinski film

Na spletnih straneh najdi podatke (osebe, datum lokacije ...) za enega od filmov ali igrico, narejeno z Blenderjem za posamezne tipične faze filmske produkcije:

- Scenarij (snemalna knjiga - označuje besedilno predlogo za snemanje filma).
- Financiranje filma (se nanaša na zbiranje kapitala za produkcijo filma; faza financiranje v filmski produkciji se mora začeti vzporedno z zbiranjem in razvojem filmskega materiala).
- Preprodukcija (vsebuje vse organizacijske in tehnične ukrepe, ki jih je potrebo narediti pred dejanskim snemanjem. To vključuje oblikovanje (angl. storyboard) in proizvodnjo scenskih slik, izbor lokacij in filmskih motivov (angl. *location scouting*)) in izbor igralcev in ostalih akterjev (angl. casting) ... V tem procesu so vključene glavne osebe filmske produkcije, ko je nekdo režiser in producent filma).
- Snemanje (predstavlja izvedbeno fazo filmske produkcije).
- Postprodukcija (angl. *post production*, vključuje vse korake za obdelavo posnetkov po snemanju v filmu, televiziji in fotografije).
- Distribucija in filmski prihodki (samo vrednotenje filma) označuje vse možne oblike prihodkov za film. Osnovna ideja izkoriščanja in predvajanja filma je od prodaje filmskih pravic za širšo uporabo samega filmskega izdelka).

Več o filmu in filmski umetnosti je na naslednjih spletnih povezavah:

<http://de.wikipedia.org/wiki/Film>, 17. 6. 2012

<http://de.wikipedia.org/wiki/Filmgenre>, 17. 6. 2012

<http://en.wikipedia.org/wiki/Film>, 17. 6. 2012
http://en.wikipedia.org/wiki/Film_genre, 17. 6. 2012

Medpredmetne povezave:

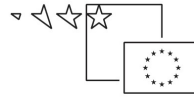
- Slovenščina
- Informatika
- Oblikovanje
- Avtomatizacija

Učna situacija 3: Računalniška grafika

V tem poglavju bo naprej podan krajši pregled o slikah, njihovih gradnikih, nastanku barv, barvnih globinah in rasterizaciji. Širše bo predstavljen vektorski zapis SVG (Scalable Vector Graphics) in in pogled na črte in krivulje v ravnini in 3D-prostoru. Skratka tiste temeljne osnove, ki so potrebne za osnovno razumevanje računalniške grafike.

Beseda grafika ima grški izvor v besedi γραφικός (graphikos) in je vizualna predstavitev (slika 78) na neki površini, kot je stena, platno, zaslon, papir ali kamen, ki obvešča, prikazuje ali je namenjena zabavi. Sicer beseda grafika izhaja iz besede graf (veja matematike, ki se ukvarja z risanjem funkcij).

Sodeč po Wikipediji (<http://sl.wikipedia.org/wiki/Grafika>, 10. 6. 2012) je pojem **grafika** ena od vej **likovne umetnosti**. Pri izdelavi grafike mora umetnik najprej s posebnim delovnim postopkom izdelati **matrico** v nemem materialu, kot npr. lesu, plastični masi, kamnu ali kovini. Matrica je izdelana v negativu, po obarvanju s črnilom ali barvo, s pritiskom na papir, dobimo odtis v pozitivu, to je grafika. Matrice v lesu so **lesorezi**, v kamnu - **litorezi**, v bakru - **bakrorezi** itd. Kvaliteta uporabljenih materialov odloča o tem, koliko dobrih odtisov grafik (v pozitivu) bomo lahko naredili z isto matrico. Odtise umetniki oštevilčijo s števili od 1 do 50 pri lesorezih, od 1 do 200 pri litografijah in od 1 do 500 pri bakrorezih.



Slika 78: Gora Fuji, od šestintrideset pogledov na goro Fuji, barvni lesorez avtorja Katsushika Hokusai, vir: <http://en.wikipedia.org/wiki/Printmaking>, 10. 6. 2012

Ločimo več vrst grafik, in sicer:

- reproduktivna ali industrijska grafika,
- umetniška ali originalna grafika.
- računalniška grafika (slika 79).



Slika 79: Virtualna resničnost v Second Life, kjer so avatarji (virtualna 3D-telesa obiskovalcev) prisotni na predavanju

Več o grafiki najdete na nemški ali angleški wikipediji s spletnim naslovom:

<http://de.wikipedia.org/wiki/Grafik>, 6. 7. 2012,

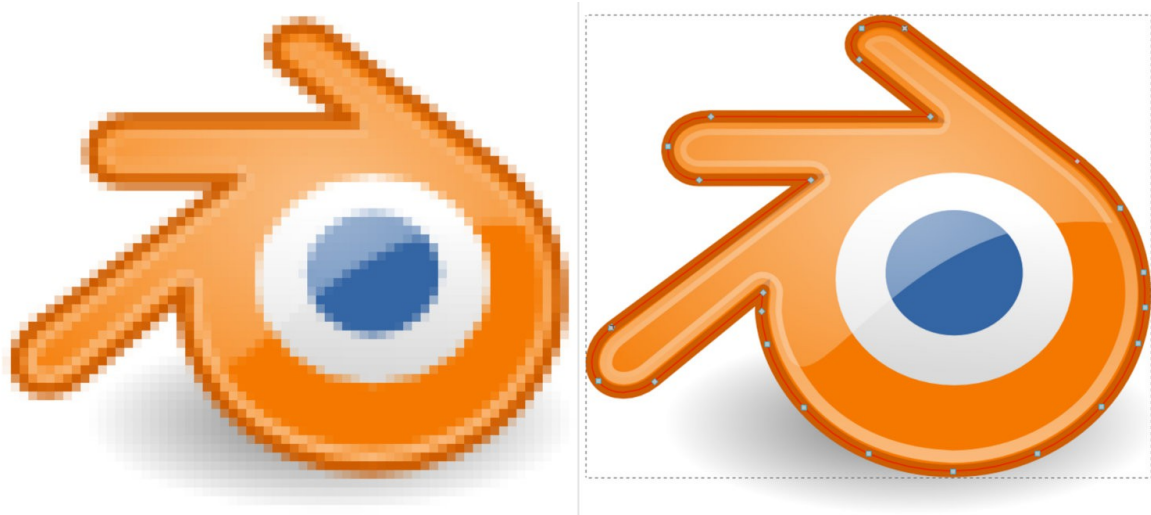
<http://en.wikipedia.org/wiki/Graphics>, 6. 7. 2012.

Nas v tem gradivu bo obravnavana predvsem računalniška grafika oz. z računalnikom ustvarjene slike, ki naj bi bile čim bolj podobne resničnim slikam, kot jih npr. naredimo z digitalno kamero.

Od 3D-točke do upodobitve

Digitalna slika lahko nastane z upodabljanjem (računanjem vrednosti) ali z digitalno kamero in je dvodimenzionalni prikaz ali zapis slike s pomočjo končne množice digitalnih vrednosti najmanjših gradnikov, ki so pri rastrskih slikah slikovne točke (angl. Pixels) (leva slika 80) in pri vektorskih slikah so matematične formule (parametrični opis formul in/ali krivulj - desna slika

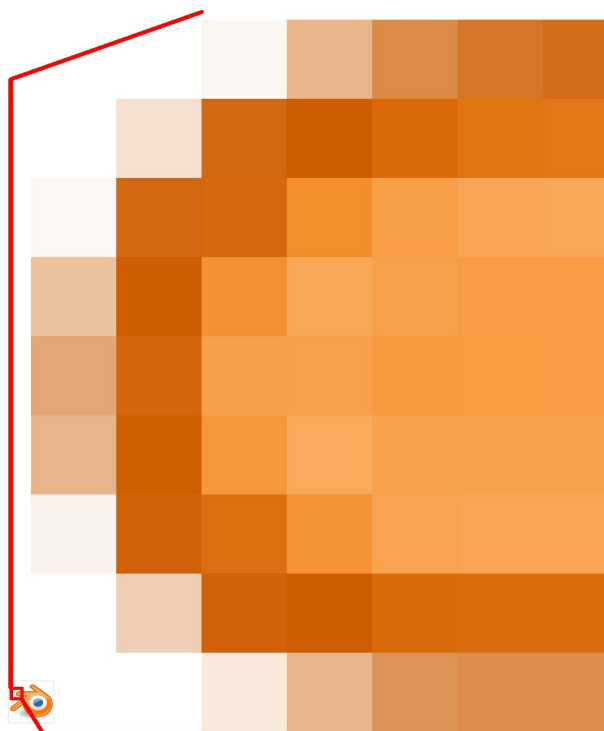
80). Digitalne slike nastanejo razen iz digitalne kamere (fotoaparata) še s pomočjo optičnega bralnika (skenerja), zračnega radarja, s seizmološkim profiliranjem, 3D-optičnim branjem, z računalniškim risanjem, modeliranjem in konstruiranjem ... Digitalne slike (grafiko) delimo na rastrske in vektorske (slika 80).



Slika 80: Rastrska in vektorska grafika logotipa programa Blender

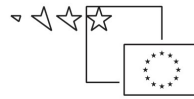
Na prejšnji sliki se lepo vidi razlika med bitnimi in vektorskimi slikami. Vektorsko sliko lahko poljubno skaliramo (večamo ali manjšamo), medtem ko rastrske (nepravilno se pogosto uporablja tudi izraz bitna) ne moremo.

Rastrska slika je sestavljena iz slikovnih točk (tipično se pridobi z digitalno kamero ali z računalniškim upodabljanjem - slika 81). Na rastrskih slikah, ki imajo poševne črte, se na sliki po skaliranju pojavljajo nazobčani robovi oz. »stopnice«. Da se izognemo tem pojavom, se uporabi antialias (angl. anti-aliasing) oz. sosednje slikovne točke se pobarvajo s to barvo in s sosednjo barvo okolice, dobimo prelivanje barv (s tem »stopničenje« ni več vidno).



o sofinanciranje

Slika 81: Rastrska slika je sestavljena iz slikovnih točk



Vektorska grafika je sestavljena iz matematičnih izrazov in/ali krivulj. Primerna je za tehnične skice, risbe in modeliranje ter 2D- in 3D-animacije (slika 82).



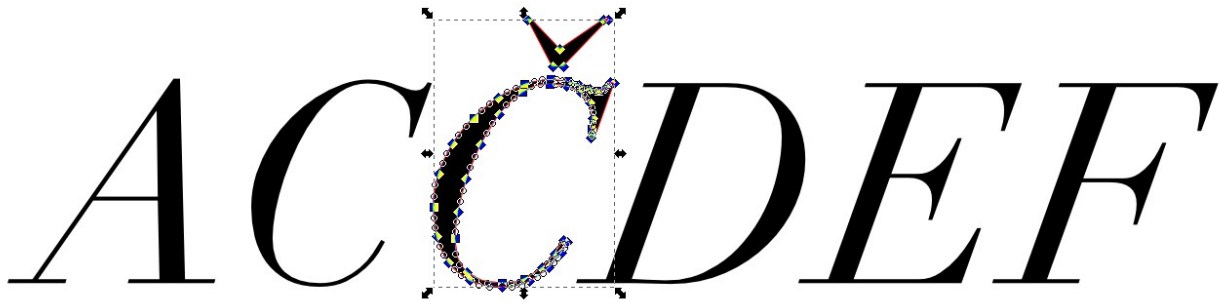
Slika 82: Vektorska risba človeka in besedila, ki sledi krivulji

Vektorske risbe srečamo na računalniku, ko beremo črke oz. besedila, ki so sestavljena iz njih. Pisava oz. vsaka črka (ali drugi znak) oz. glif je »prava« risba, ki je nastala pod avtorskim peresom in se pozneje shrani

58

Učno gradivo je nastalo v okviru projekta Munus 2. Njegovo izdajo je omogočilo sofinanciranje Evropskega socialnega sklada Evropske unije in Ministrstva za šolstvo in šport.

kot znak v datoteko s pisavo. Te imajo večinoma končnico OTF, TTF ali PS1 (slika 83).



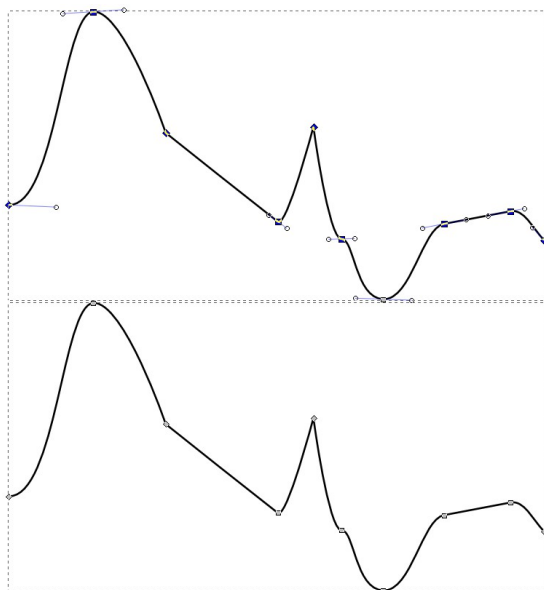
Slika 83: Vse pisave oz. tipografije so tudi »prave« vektorske risbe

Večina vektorskih programov uporablja za zapisovanje točk in črt matematični vektorski zapis in/ali t. i. krivulje Bézier ter tudi druge tipe krivulj in površin.

Krivulje Bézier

Krivulje Bézier [6] je uporabil francoski matematik Pierre Étienne Bézier leta 1962 pri oblikovanju avtomobilskih karoserij pri Renaultu. Te parametrične krivulje so pomembne v računalniški grafiki za ustvarjanje vektorskih slik. Krivuljo določajo štiri točke: začetni in končni položaj ter dve ločeni vmesni kontrolni točki (angl. Control points). Skrajne točke na krivulji so vozlišča (angl. Knots) in morebitne vmesne točke, ki vplivajo težnostno na krivulje se imenujejo kontrolne točke.

Bézier-jeve predmete lahko spreminjamo s premikanjem teh točk ali ročic s pomočjo miške (slika 84).



Slika 84: Krivulja, definirana z vozlišči in kontrolnimi ročaji (zgoraj) in samo z vozlišči (spodaj)

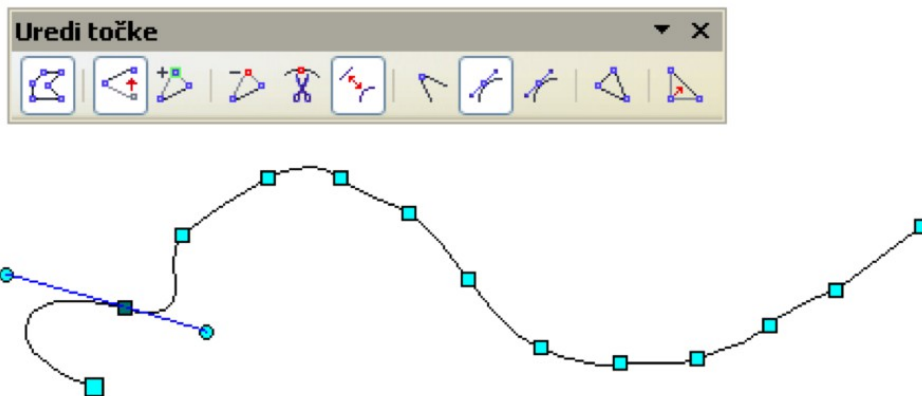
Matematično so te krivulje izražene s polinomi prve, druge in tretje stopnje:

$$B(t)=(1-t)P_0, t \in [0,1]$$

$$B(t)=(1-t)^2P_0+2t(1-t)P_1+t^2P_2, t \in [0,1]$$

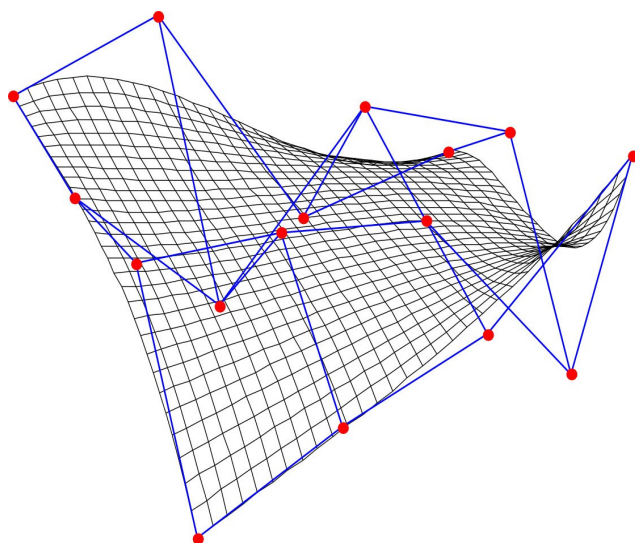
$B(t)=(1-t)^3P_0+3t(1-t)^2P_1+3t^2(1-t)P_2+t^3P_3, t \in [0,1]$, kjer je parameter t del uniformnega prostora in se lahko giblje v intervalu $[0, \dots, 1]$ in s črkami P_0, P_1, P_2, P_3 so poimenovane kontrolne točke.

Krivulja Bézier vsebuje **voziščne** točke in vsaka ima dve kontrolni ročici (slika 85) iz programa Libre Office Writer.



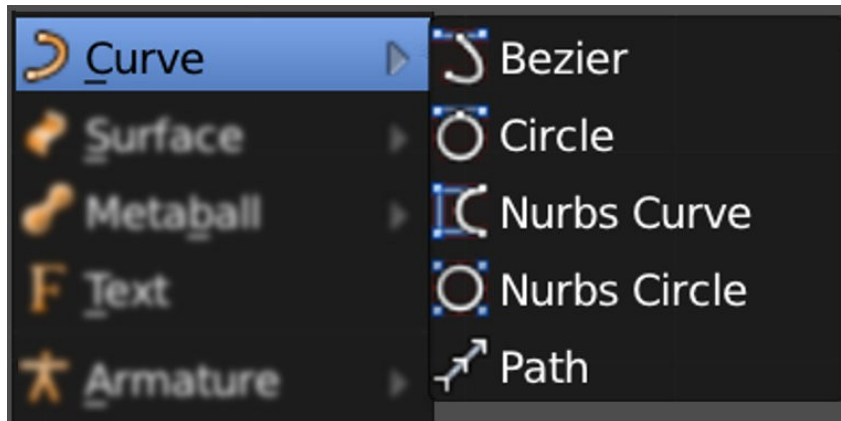
Slika 85: Krivulja Bézier v načinu urejanja z dvema kontrolnima ročicama

Posplošitve Bézierovih krivulj na višje razsežnosti se imenujejo Bézierove površine (slika 86). Tu so kontrolne točke in krivulje so izven Bézierove površine. Te so identične le za vogalne točke B. površine oz. robne točke B. krivulje.



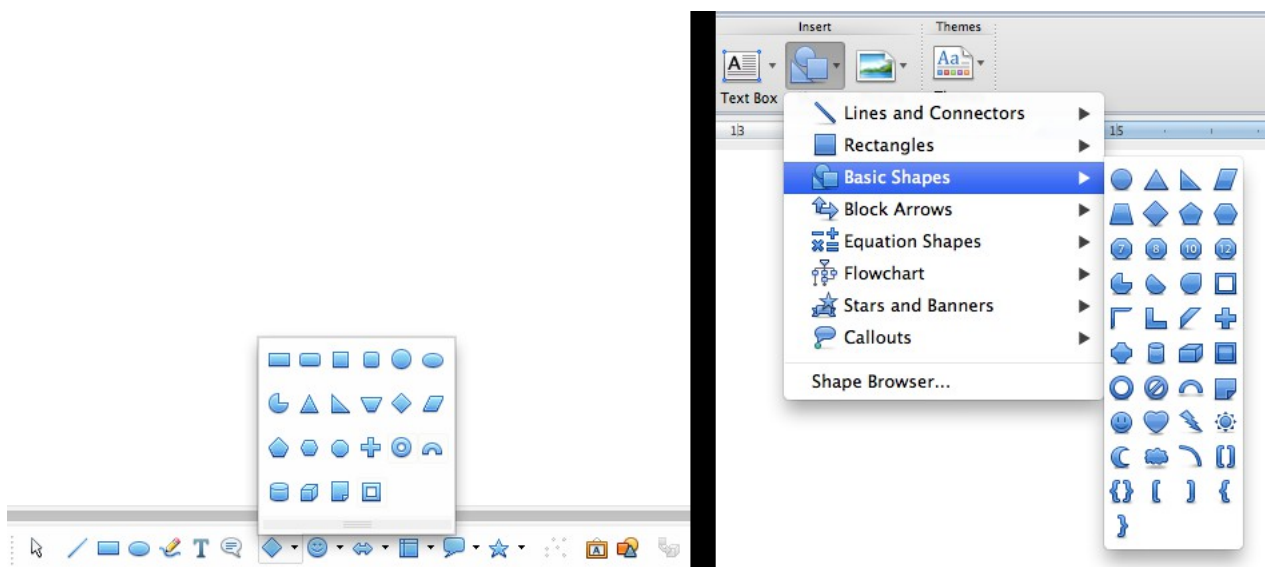
Slika 86: Bézierova površina (rdeča so kontrolne točke, modra kontrolni robovi in črna mreža predstavlja aproksimacijo površja)

Blender vsebuje številne tipe različnih Bézier- in NURBS-krivulj in NURBS-površin kot osnovne elemente oz. izhodišče za nadaljnje modeliranje lastnih modelov (slika 101). V tem meniju je pod oznako Bézier po vstavljanju vidna odprta pot z dvema kontrolnima točkama, ki imata dve ročici. Pod oznako Circle po vstavljanju v sceno je viden Bézier-krog s štirimi kontrolnimi točkami, kjer ima vsaka dve kontrolni ročici.



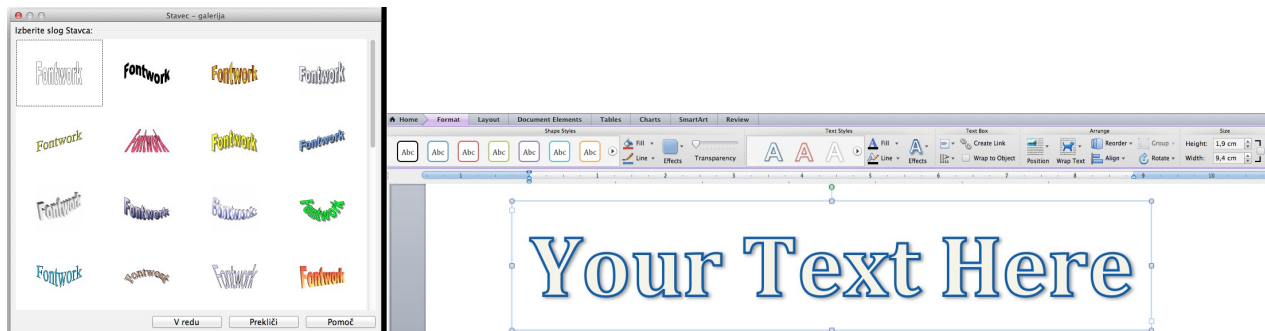
Slika 87: Meni za vstavljanje krivulj angl. Curve

Pisarniški programi, kot sta np. Libre office Writer ali MS Word, imajo vgrajen lasten podprogram za preprosto vektorsko risanje (slika 88).



Slika 88: Orodja za risanje v Libre Office Writer 3.5 (levo) in MS Word 2010 (desno)

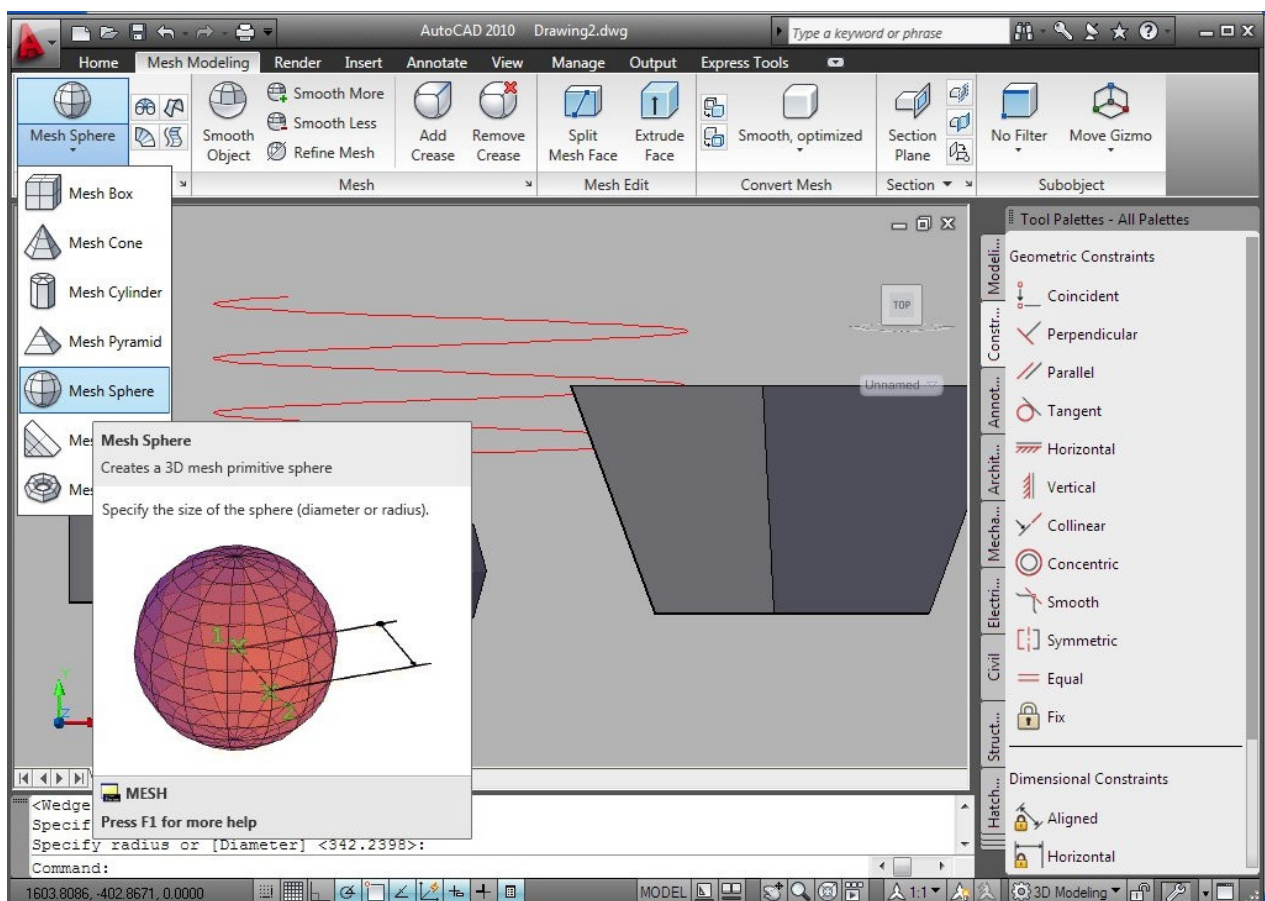
Prav tako imata Libre Office Writer in MS Word podprogram za izdelavo naslovnih napisov (slika 89).



Slika 89: Stavec v Libre Office 3.5 (levo) in Word Art v MS Wordu 2010 (desno)

http://en.wikipedia.org/wiki/B%C3%A9zier_curve
http://en.wikipedia.org/wiki/B%C3%A9zier_surface

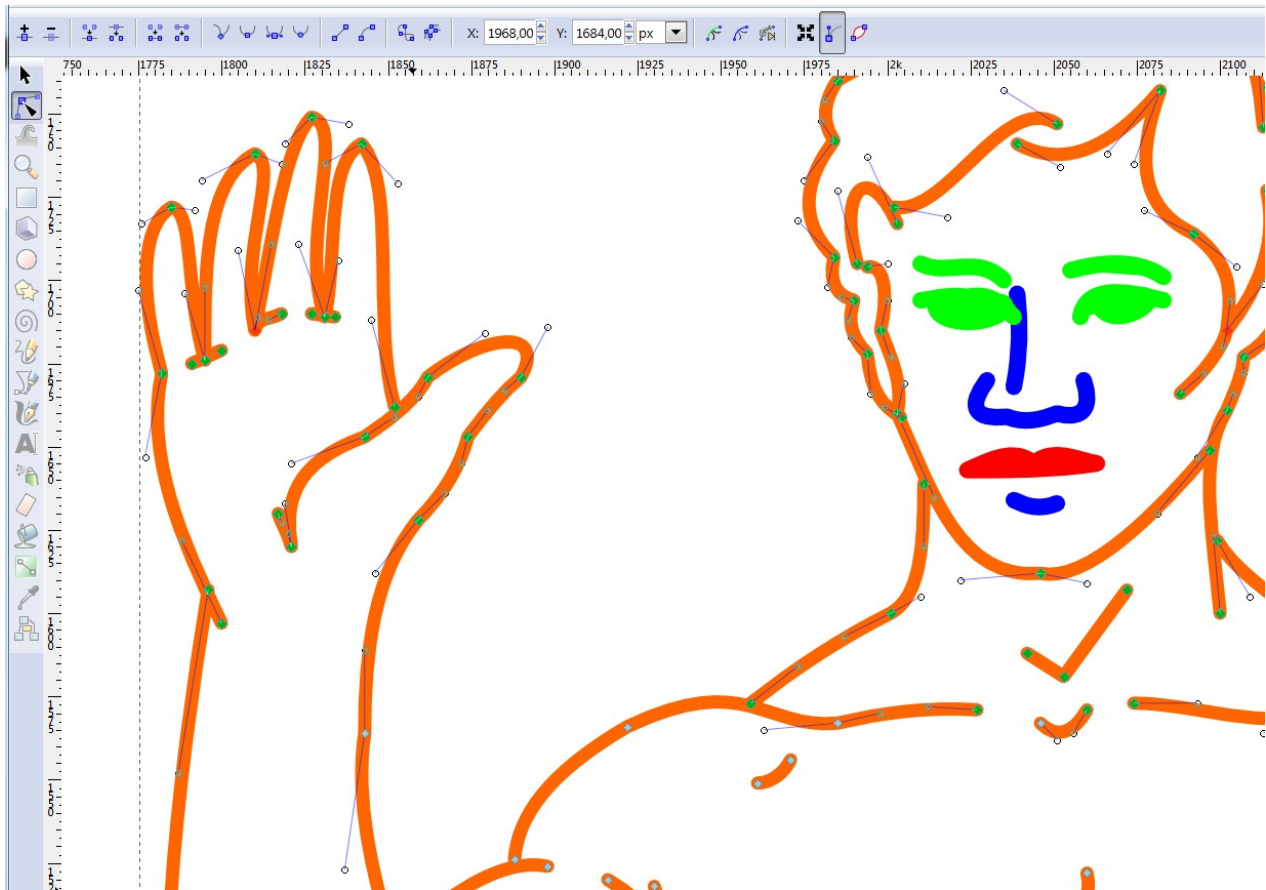
Vsi omenjeni programi vključno z risarjem (angl. Draw) so namenjeni za izdelavo bolj preprostih risb. Za profesionalne načrte za strojništvo, gradbeništvo in arhitekturo se veliko uporablja program AutoCAD (slika 90).



Slika 90: Pogled na uporabniški vmesnik programa AutoCAD 2010

Za risanje vektorskih risb je zelo popularen odprtokodni program Inkscape (<http://inkscape.org/>),

ki v osnovi shranjuje v formatu SVG (slika 91), ki je bil razvit in normiran s strani W3-konzorcija. Ta format se veliko uporablja za vse vrste skic in risb na spletu npr. v največji odprti spletni enciklopediji Wikipediji.



Slika 91: Vidna so vozlišča in kontrolne ročice v vektorskem programu Inkscape 0.48

Zapis SVG, ki ga uporablja tudi program Inkscape, zapisuje risbo s besedilnimi znaki ASCII oz. temu uradno pravimo skladno z razširjenim označevalnim jezikom XML (eXtensible Markup Language -). Poglejmo nekaj več informacij o SVG iz wikipedije http://sl.wikipedia.org/wiki/Scalable_Vector_Graphics, 10. 6. 2012.



S spletne strani <http://inkscape.org/> prenesite zadnjo različico odprtokodnega programa Inkscape in z orodjem Risanje krivulj in ravnih črt narišite naslednjo sliko 92 (predlagam, da se tisti, ki ste bolj zagnani, preizkusite v risanju slike 82 ali slike 91), zraven se podpišite in dodajte tudi datum. Datoteko risbe shranite pod imenom npr. ime_priimek_inkscape_1.svg. Na naslednj sliki najdete tudi ikono orodja za lažje prisovanje.




Slika 92: Orisna črta logotipa Blender

Format SVG

Scalable Vector Graphics ([kratica SVG](#), [slovensko skalirna vektorska grafika](#)) je [označevalni jezik](#) za opis dvorazsežne statične ali animirane in risane [vektorske grafike](#) v [XML](#).

Preprost primer vektorskega zapisa SVG je podam v tabeli 1, kjer je lepo vidna preprosta sintaksa razširjenega označevalnega jezika XML.

Tabela 1: Koda za risanje preproste vodoravne črte

Koda SVG	Risba
<pre><line x1="0" y1="100" x2="100" y2="0" stroke-width="2" stroke="black" /></pre>	

Septembra [2001](#) ga je [W3C](#) predlagal za rabo. Razvijal se je kar nekaj časa. Najprej sta družbi [Macromedia](#) in [Microsoft](#) predstavili jezik [VML](#), [Adobe](#) in [Sun Microsystems](#) pa konkurenčni format [PGML](#). SVG je izvirno le podpiral [spletni brskalnik Amaya](#). V drugih brskalnikih so bili dolgo časa potrebni dodatni programi, kot sta [Adobe SVG Viewer](#) ali [Corel SVG Viewer](#). Sedanja različica brskalnika [Mozilla](#) z imenom »Croczilla«
sedaj podpira dele standarda W3C za SVG, veliko stvari pa še ne, vendar želijo doseči, da bi bil brskalnik sposoben prikazovati SVG brez dodatkov. Spletni brskalnik [Konqueror](#) projekta [KDE](#) ima zadovoljivo in dovršeno podporo za SVG z imenom [ksvg](#), ki jo bo v prihodnosti verjetno prevzel [Safari \(brskalnik\)](#) družbe [Apple Computer](#). [Javanski](#) programi lahko za prikaz, izdelavo in prirejanje grafike SVG uporabljajo orodni pribor [Batik SVG Toolkit](#).

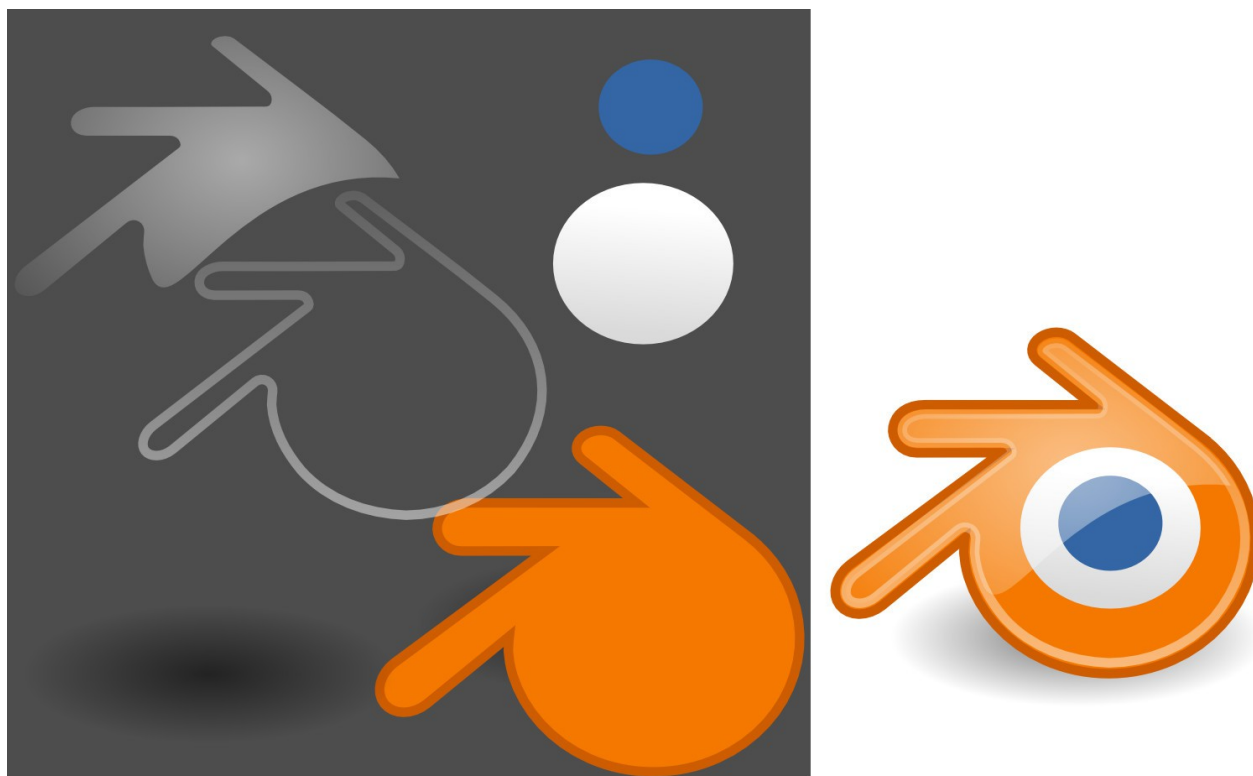
Konzorcij W3C je izdal priporočila za SVG verzijo 1.0, 1.1 in 1.2 (lahko za odjemalce), del te kode in logotip je viden na sliki 93.

```
<?xml version="1.0" en
<svg version="1.0" xi
<defs>
  <linearGradient x1="99.7"
</defs>
  <use xlink:href="#box_gr
  <use xlink:href="#circle
  <use xlink:href="#circle
  <line x1="100" y1="300"
  <!--add more con
  <circle cx1="90"
</svg>
```



Slika 93: Ponazorjen logotip in del kode SVG-značk

V vektorskih programih se risbe izdelujejo s pomočjo osnovnih oblik, ki imajo lastnost barve črte (vrsta in debeline črte) in vrsto teksture ali barve polnila notranjosti lika. S pomočjo izbora ustreznega vrstnega reda nalaganja posameznih delov risbe in z določanjem prosojnosti in prekrivnosti se lahko doseže poljuben učinek vključno s podobo 3D-predmeta (slika 94).



Slika 94: Logotip Blenderja je sestavljen iz številnih risb, ki so združene v končno podobo

SVG se še vedno razvija. Njegov razvoj so spodbujali proizvajalci vtičnikov, zdaj izvorni zapis SVG izvajajo vsi najbolj sodobni brskalniki. Še vedno traja postopek optimiziranja za še tesnejše povezovanje s HTML in CSS v brskalniku. Specifikacijo za SVG Tiny 1.2 so razvili in uvedli proizvajalci in prodajalci mobilnih naprav in ta ima vključenih nekaj več funkcij, primernih za te naprave, kot pa je običajno pri spletnih brskalnika za namizne računalnike. Več o razvoju in nadaljevanju tega standarda je na spletišču W3C-konzorcija:

<http://www.w3.org/Graphics/SVG/WG/wiki/Roadmap>, 10. 6. 2012.

Končnici slikovne datoteke sta .svg ali .svgz in je MIME-Tip (image/svg+xml). Razvit je bil s strani W3-konzorcija in prav tako normiran kot: HTML, CSS, XML in PNG. Neodvisen od platforme oz. operacijskega sistema, odprto dokumentiran in prosto uporaben.

Format SVG je vektorsko orientiran in ne določa slikovnih točk temveč objekte, ki imajo svoje lastnosti, kot so: položaj, debelina črte, polnilo (preliv ali rastrska slika). Tak zapis je idealen za vse vrste tehnične grafike, za ilustracije in konstrukcijsko grafiko.

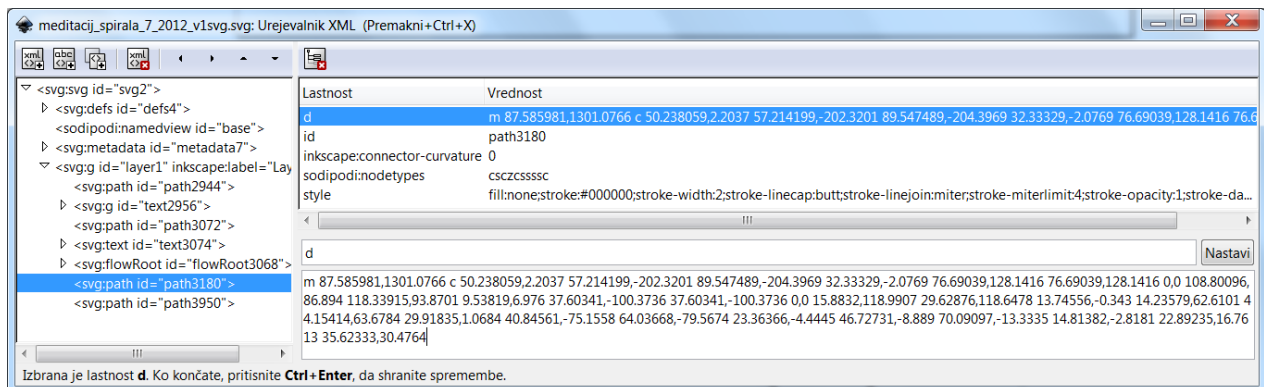
SVG-format ima naslednje lastnosti:

- Nastal na osnovi XML-a: SVG je osnovan na jeziku XML. Ta format se lahko v vsakem trenutku DTD validira. Kar je še najbolj pomembno, je popolnoma besedilni format zapisa. S pomočjo dokumentnega objektnega modela (Document Object Model (DOM)) je možen direkten dinamičen dostop direktno do vsakega opisa znotraj SVG-datoteke.
- XLink-podpora: XLink (XML Linking Language) po shemi (načrt) za opis nadbesedilnih povezav (Hyperlinks) v XML-u. Tudi na SVG se lahko uporabijo te sheme. Na ta način je možno ustvariti SVG-grafike, ki so občutljive na povezave (slike s povezavami). Povezave vodijo do

lokalnih ali globalnih ciljev, ki so kjerkoli v svetu.

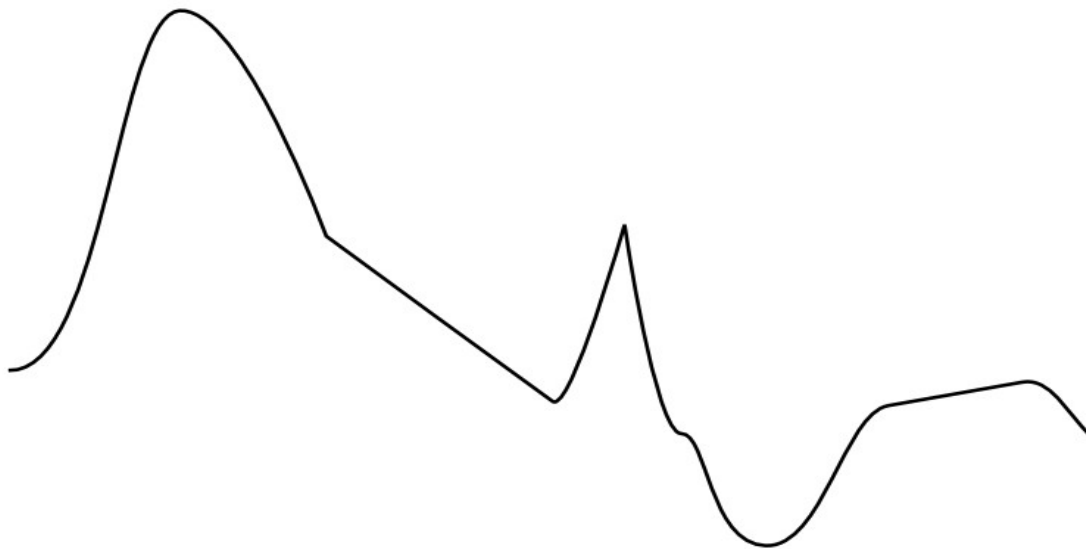
- Zoomiranje (skaliranje) - in možnost zasuka: Vektorske grafike je mogoče skalirati brez izgub. Za osebe s slabim vidom je mogoče povečati majhno grafiko. Takšne grafike je mogoče preprosto zasukati.
- SMIL-vmesnik: SMIL (angl. Synchronized Multimedia Integration Language -sinhronizacijski večpredstavnostni integracijski jezik) je prav tako normiran jezik s strani W3-konzorcija. Ta opisuje časovne trakove za večpredstavnostne (multimedijske) dogodke. Skupaj s SMIL-om je mogoče izdelati dinamične SVG-grafike z drugimi elementi, kot je npr. zvok.

Koda za spiralo na sliki 82, ki je na strani 58, je vidna na sliki 95. Na levem delu slike se vidijo z različnimi parametri za »id« posamezni tipi objektov.



Slika 95: Koda za spiralo na sliki 82

Narisana pot ima parametra M za premik (moveto) in L za črto (line). Obe oznaki sta za kubični in kvadratni krivulji Bézier. S parametrom id je opisana identifikacijska številka poti. Bolj kompleksen primer SVG-krivulje z različnimi lastnostmi prehoda krivulje okrog kontrolnih točk (na nasprotnih straneh kontrolne krivulje ime ta lahko različno lastnost- npr. oster prehod ali bolj počasen upad) je viden na sliki 96. Vrsta prehodov kontrolnih točk je podana s parametrom sodipodi:nodetypes.



Slika 96: Kompleksna krivulja z različnimi lastnostmi prehoda krivulje okrog vozlišč (mehki in osti prehodi)

Koda kompleksna krivulja z različnimi lastnostmi prehoda krivulje okrog vozlišč je vidna v tabeli 2. Prehodi vozlišča (angl. Node oz. Knot) imajo lahko eno od naslednjih štiri lastnosti: smooth (s), corner (c), symmetric (z) in [auto-smooth](#) (a) (v prevodu: mehkejše (s), oglišče (c), simetrično (z) in samodejno mehkejšo (a)). Črke v oklepaju za imenom tipa vozlišča predstavljajo kratico za posamezen tip vozlišča. Vsak objekt (predmet) ima lastno identifikacijsko oznako (id), ki se prične z oznako path in ki ji sledi številka. Pod lastnost slog (angl. Style) je zapisana lastnost črte in polnila.

Tabela 2: Zapis kode za zgornjo kompleksno krivuljo

Lastnost	Vrednost
d	m 87.585981,1301.0766 c 50.238059,2.2037 57.214199,-202.3201 89.547489,-204.3969 32.33329,-2.0769 76.69039,128.1416 76.69039,128.1416 0,0 108.80096,86.894 118.33915,93.8701 9.53819,6.976 37.60341,-100.3736 37.60341,-100.3736 0,0 15.8832,118.9907 29.62876,118.6478 13.74556,-0.343 14.23579,62.6101 44.15414,63.6784 29.91835,1.0684 40.84561,-75.1558 64.03668,-79.5674 23.36366,-4.4445 46.72731,-8.889 70.09097,-13.3335 14.81382,-2.8181 22.89235,16.7613 35.62333,30.4764
id	path3180
inkscape:connector-curvature	0
sodipodi:nodetypes	csczcssscc
style	fill:none;stroke:#000000;stroke-width:2;stroke-linecap:butt;stroke-linejoin:miter;stroke-miterlimit:4;stroke-opacity:1;stroke-

	dasharray:none
--	----------------

Format SVG je pri delu z Blenderjem pomemben, ker lahko te krivulje vanj direktno uvozimo in jih tako kar naprej uporabljamo.

Profili v formatu SVG

V SVG obstajajo tri vrste profilov, ki so namenjene za prikaz SVG-slike na različnih izhodnih napravah in glede na zmogljivost naprave so za vsako od teh določeni različno obsežni SVG-standardi. Ti se podajo v profilu korenkega elementa (SVG) z atributom *baseProfile*.

SVGT

(Base profil = "*tiny*" oz. "majhen") je zelo omejen in namenjen za mobilne internetne naprave (MID- [Mobile internet devices](#)) in mobilne telefone. Določa le majhno podmnožico v SVG-standardu.

SVGB

(Base profil = "*basic*" oz. "osnovni" v prevodu "enostavno, osnovno") je namenjen za naprednejše mobilne internetne naprave, vendar ne določa celotnega SVG-standarda.

SVG F

(Base profil = "*full*" oz. "polno" angleško "popolnoma, popolnoma") je namenjen za računalnike kot izhodne naprave in v celoti podpira standard SVG.

Z uporabo profilov SVG je mogoče, da lahko standardne datoteke zadovoljivo uporabljajo SVG tudi na napravah z malimi strojnimi zmogljivostim (npr. dlančniki in mobilni telefoni).

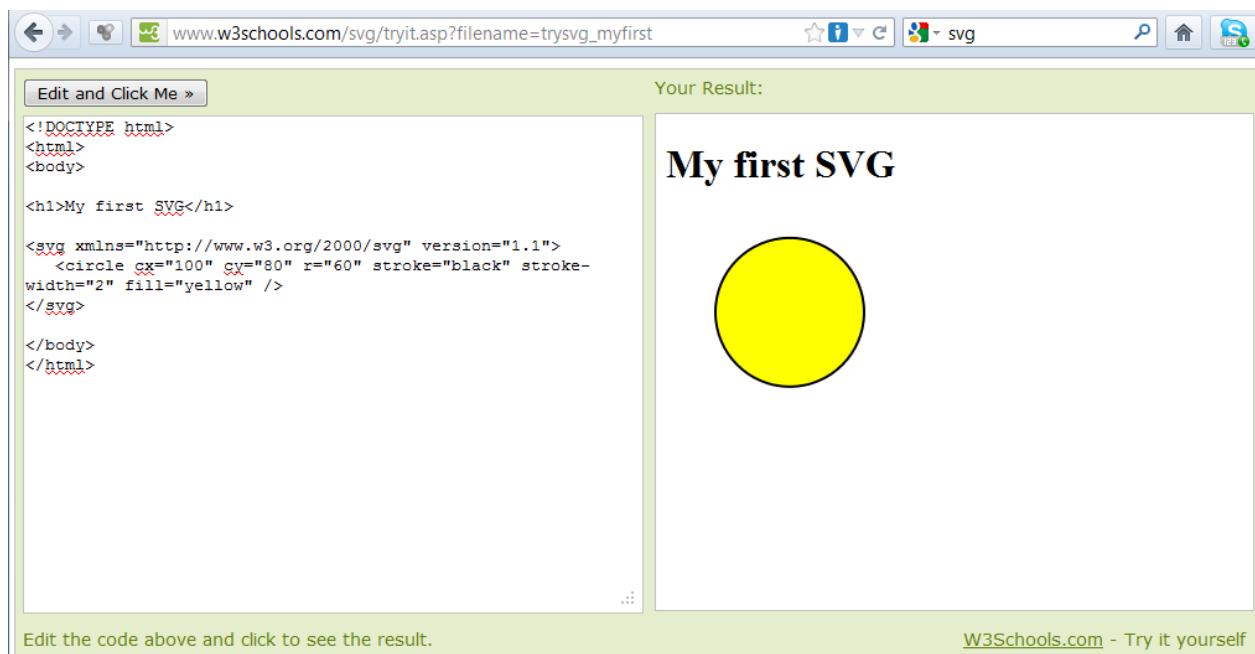
Več o SVG-zapisu najdete na spletnih naslovih:

- http://sl.wikipedia.org/wiki/Scalable_Vector_Graphics, 10. 6. 2012
- http://de.wikipedia.org/wiki/Scalable_Vector_Graphics, 10. 6. 2012
- http://en.wikipedia.org/wiki/Scalable_Vector_Graphics, 10. 6. 2012
- <http://www.w3.org/Graphics/SVG/>, 10. 6. 2012
- <http://www.w3schools.com/svg/default.asp>, 10. 6. 2012
- <http://www.adobe.com/svg/>, 10. 6. 2012
- <http://www.fileinfo.com/extension/svg>, 10. 6. 2012
- http://de.wikipedia.org/wiki/Scalable_Vector_Graphics, 10. 6. 2012



Pojdite na spletno stran <http://www.w3schools.com/svg/default.asp> in zamenjajte krog (slika 97) v rdečo, zeleno in modro (lahko tudi v svojo najljubšo varno spletno barvo), zraven se podpišite (dodajte tudi datum) ter zmanjšajte in povečajte krog na največjo oz. najmanjšo mero in si te

vrednosti zapišite. Eno od kod kopirajte v nek urejevalnik HTML-kode, datoteko shranite in s spletnim brskalnikom si oglejte vašo risbo. Za vsako večjo spremembo naredite zaslonsko sliko (Print Screen), ki jo boste pozneje oddali v spletno učilnico ali jo prilepili v zvezek.



Slika 97: Primer spletne strani <http://www.w3schools.com>, kjer lahko interaktivno s kodo igramo na spletni strani v spletnem brskalniku



1. Preglejte z vsaj 4 različnimi spletnimi brskalniki (npr. Mozilla Firefox, Safari, Chrome, Opera in Internet explorer) primere kode SVG, ki so dostopni na spletišču:

<http://www.w3.org/Graphics/SVG/IG/resources/svgprimer.html>. Za posamezno kodo napišite ugotovitve ali posebnosti glede na uporabljen spletni brskalnik.

2. Z vsaj 4 različnimi spletnimi brskalniki (npr. Mozilla Firefox, Safari, Chrome, Opera in Internet explorer) si oglejte SVG animirano grafiko trojanskega stolpa in v kodi ugotovite, kako je ta animirana (kopirajte ustrezno kodo in jo komentirajte):

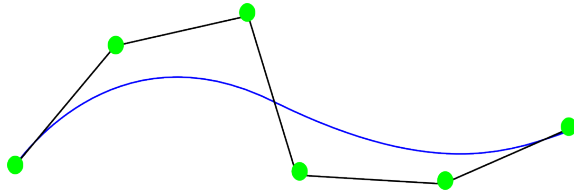
<http://upload.wikimedia.org/wikipedia/commons/6/6c/Trajans-Column-lower-animated.svg>.

3. Z vsaj 4 različnimi spletnimi brskalniki (npr. Mozilla Firefox, Safari, Chrome, Opera in Internet explorer) si oglejte SVG animirano grafiko vrteče nogometne žoge in v kodi ugotovite, kako je ta animirana. Kopirajte ustrezno kodo in jo komentirajte:

http://upload.wikimedia.org/wikipedia/commons/thumb/4/4f/Soccer_ball_animated.svg/220px-Soccer_ball_animated.svg.png.

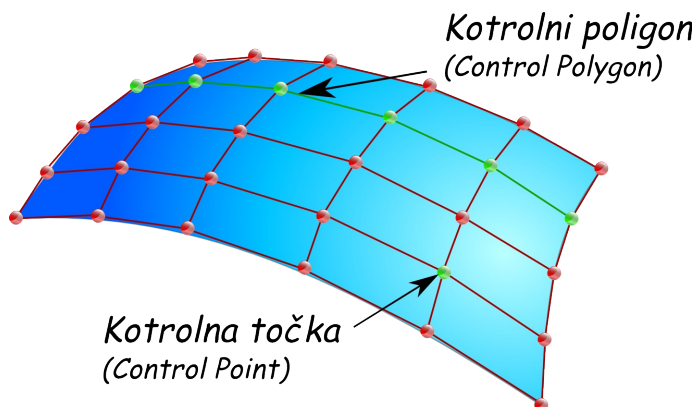
NURBS-krivulje

Kratica [6] NURBS-krivulja (slika 98) je izpeljana iz angleške besedne zveze »Non Uniform Rational Basis Spline«, kar se prevaja kot neuniformne racionalne B-krivulje, še bolj preprost prevod je »neenakomerna racionalna osnovna krivulja«.



Slika 98: NURBS-krivulja s šestimi kontrolnimi točkami

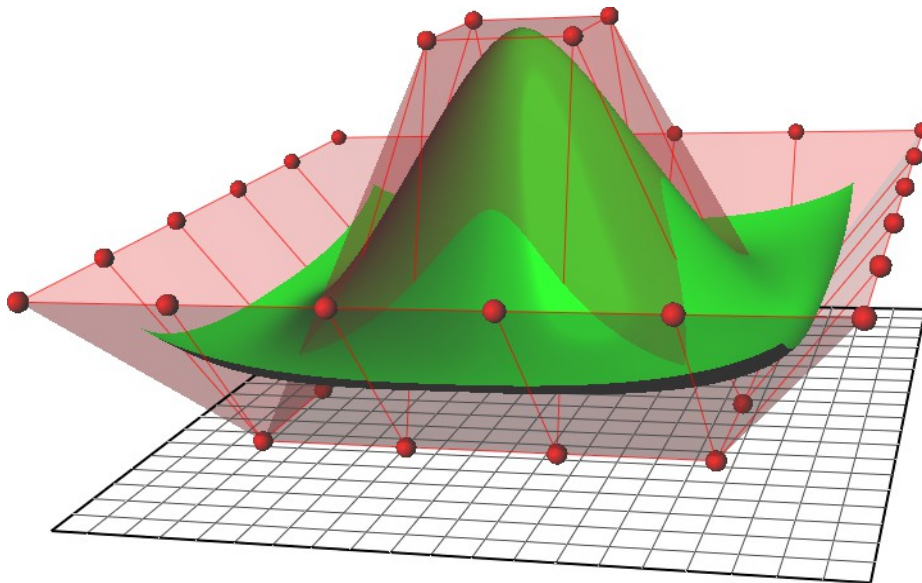
NURBS-krivulje so matematične krivulje in osnova NURBS-ploskev (slika 99), ki se uporabljajo v področju računalniške grafike, na primer v CGI (angl. Computer Generated Imagery), CAD (angl. Computer Aided Design) ali za modeliranje poljubnih oblik. Predstavitev geometrijskih podatkov se izvede preko nesistematično funkcionalno določene geometrije elementov. Kontrolne točke niso nujno na sami krivulji oz. ploskvi, temveč so praviloma izven njih in so »težnostno silo« povezane s ploskvijo (kot z nekakšnimi nevidnimi elastičnimi trakovi).



Slika 99: NURBS-ploskev

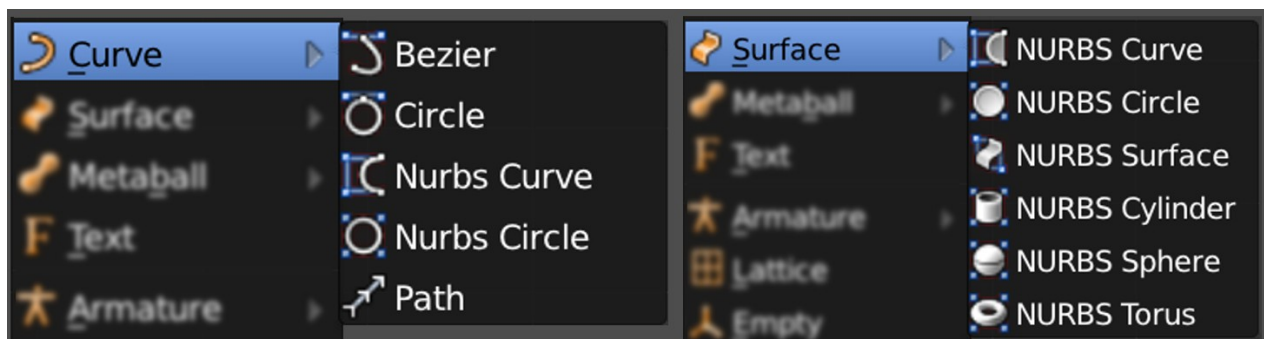
S pomočjo krivulj NURBS so načeloma lahko predstavljeni vsi tehnični izdelki ali fizične oblike (slika 100). Te oblike krivulj oz. površja se uporabljajo v grafiki, 3D-modeliranju, arhitekturi, še zlasti pa pri projektiranju letal, avtomobilov in ladij ter ostalih izdelkov sodobnega oblikovanja.

NURBS-krivulja je posplošena [Bézierova krivulja](#). Slovencem je manj znana posebna oblika teh krivulj *planicoida*, posebej razvita za opis letalnice v Planici.



Slika 100: NURBS-površina 4. stopnje (zelene barve) določa 36 kontrolnih točk (rdeče žoge) v dvodimenzionalnem območju parametrov (spodnja mreža)

Blender vsebuje številne tipe različnih Bézier- in NURBS-krivulj in NURBS-površin, ki jih ta vsebuje kot osnovne elemente za nadaljnje modeliranje lastnih modelov (slika 101).



Slika 101: Meni za vstavljanje različnih tipov krivulj in površin, ki jih vsebuje Blender

NURBS-krivulje se veliko uporabljajo tudi v 3D-knjžnici, med ostalim tudi v [Open Graphics Library \(OpenGL\)](#).

Več o krivuljah in ploskvah lahko preberete v knjigah:

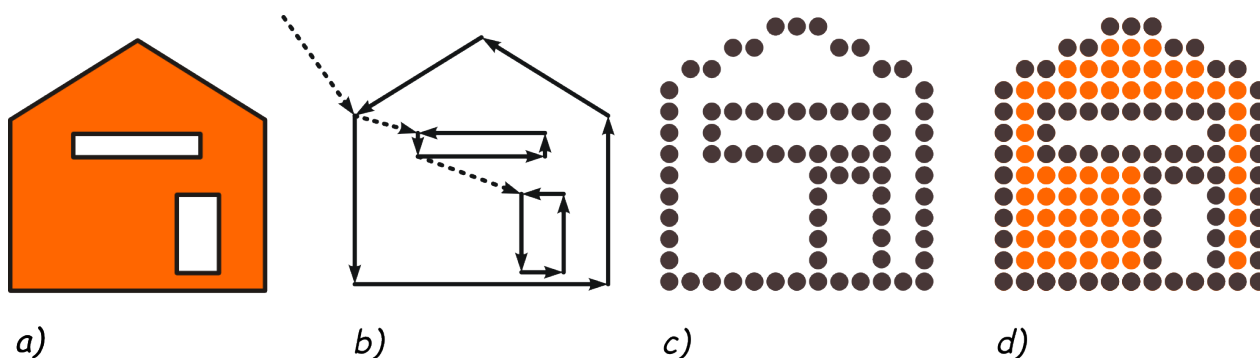
- [5] Guid, N. Računalniška grafika, Maribor:, 2001, ISBN: Založništvo fakultete za elektrotehniko, računalništvo in informatiko,
- [6] Jože Duhovnik, Milan Kljajin, Milan Opalić Inženirska grafika, Univerza v Ljubljani, Fakulteta za strojništvo 2009, ISBN 978-961-6536-31-8,
- [7] Žalik, B. Geometrijsko modeliranje 1. natis. Maribor: Fakulteta za elektrotehniko, računalništvo in informatiko, 1999. ISBN: 86-435-0297-9.

Upodabljanje risb SVG

Kot je bilo že omenjeno, se danes risbe v formatu SVG praviloma že veliko uporabljajo na spletu npr. pri spletnih straneh v Wikipediji. Preden se v spletnem brskalniku vidi vektorska grafika (slika 102 a) (slika 102 d), je potrebno enako kot za spletne strani, da se koda razčleni (izvede t.i. angl. Parsing) in potem upodobi oz. pretvori v rastrsko obliko, ki se prikaže na zaslonu. Današnji zasloni so namreč praviloma rastrski, ker so sestavljeni iz slikovnih točk (angl. Pixels).

Najlepša slika je pri priporočeni maksimalni ločljivosti zaslona.

Pri rasterizaciji, ki je lahko strojna ali programska (enako se naredi pred tiskanjem dokumenta na tiskalniku), se najprej prepozna slika po konturah (slika 102 b) in glede na gostoto rastra določi, katera slikovna točka se uporabi oz. izriše za zunanji obris objekta npr. hiše (slika 102 c). Na koncu se z barvo pobarva notranjost hiše skladno z originalnim predlogom (slika 102 d).



Slika 102: Rasterizacija vektorskega zapisa slike hiše, vir: [izboljšana slika iz http://de.wikipedia.org/wiki/HTML-Rendering](http://de.wikipedia.org/wiki/HTML-Rendering), 10. 6. 2012.

Spletni brskalnik pred upodabljanjem slike na zaslonu (slika 102 d) vso kodo, ki jo dobi od spletnega brskalnika, razčleni (angl. parsing) in na osnovi algoritma za upodabljanje le-tega prikazuje kodo bolj ali manj skladno s standardom. Za upodabljanje slike na zaslon spletni brskalniki uporabljajo pogon za upodabljanje (angl. Rendering-Engine). Programerji spletnih brskalnikov časovno več let ne sledijo standardom oz. priporočilom w3C. Tudi če s spleta prenesemo zadnjo različico spletnega brskalnika, ta praviloma v celoti ne podpira SVG in drugih standardov konzorcija W3C (tabela 3). Iz naslednje tabele je razvidno, da ima Trident pogon za upodabljanje, ki se uporablja pri spletnem brskalniku Internet Explorer 9.0, najslabšo podporo za prikaz vektorskega standarda SVG 1.1.

Tabela 3: Pogoni spletnih brskalnikov za upodabljanje in deležem podpore glede na standard SVG 1.1

Pogon za upodabljanje (angl. Rendering-Engine)	Spletni brskalnik	Podpora SVG 1.1
Presto	Opera 11.01	95,44 %
WebKit	Safari 5.0	82,48 %
	Google Chrome 10.0	89,23 %
Gecko	Mozilla Firefox 4.0	82,30 %

Trident	Internet Explorer 9.0	59,64 %
	Internet Explorer do različice 8.0	Ni (le s pomočjo vtičnikov)

Več o upodabljanju spletnih brskalnikov najdete na spletnih naslovih:
http://en.wikipedia.org/wiki/Web_browser_engine, 10. 6. 2012.
<http://de.wikipedia.org/wiki/HTML-Rendering>, 10. 6. 2012.

Povzetek uvoda v rastrsko in vektorsko grafiko

Slikovne točko so najmanjši gradniki digitalnih slik. Le-te lahko primerjamo s kamni v mozaiku, ki ima kamene, porcelanaste oz. steklene ploščice (pogosto so pri pravem mozaiku velikosti in oblika ploščic različne).

Druga vsakdanja primerjava slikovnih točk je z blagom oz. gobelinom, ker so prav tako lepo vidijo "očesa".

Obstajajo različni prosti, odprtokodni ali licenčno plačljivi programi za urejanje rastrskih in vektorskih slik, nekatere pomembnejše ikone teh programov vidimo na sliki 103.

Odličen odprtokodni program za urejanje rastrskih slik je GIMP (GNU Image Manipulation Program)(<http://gimp.org/>). Z njim je možno obdelati fotografije, risati, izdelati poljubne fotomontaže, 2D-animacije, fraktale ... Plačljiv program, podoben GIMP-u, je Adobeov Photoshop (www.adobe.com).

Inkscape (<http://www.inkscape.org>) je odprtokodni program, namenjen za vektorsko risanje v formatu SVG. SVG ("skalarna vektorska grafika") je standardiziran slikovni format za 2D vektorske slike. Plačljiv program za vektorsko risanje je Adobe Illustrator (www.adobe.com). Adobe Flash je program za 2D-animacije. Autodesk Maya (<http://www.autodesk.com/maya>) je podoben kot Bender, le da je komercialen program za 3D-animacijo. Synfig Studio je brezplačna in odprtokodna programska oprema za 2D-animacijo, zasnovan kot močna industrijska rešitev za ustvarjanje animacije filmske kakovosti z uporabo vektorskih in rastrskih slik. Odpravlja potrebo po ustvarjanju animacije z ločenimi okvirji. S tem se omogoča izdelava 2D-animacije z višjo kakovostjo z manj ljudmi in proračunskimi sredstvi projekta. Synfig Studio je na voljo za operacijske sisteme Windows, Linux in MacOS X (<http://www.synfig.org/cms/>).



Slika 103: Ikone grafičnih programov GIMP, Inkscape, Blender, Photoshop, Illustrator, Flash in Maya

Več o programih za urejanje rastrskih in vektorskih slik najdete na spletnih naslovih:
http://en.wikipedia.org/wiki/Autodesk_Maya, 10. 6. 2012
<http://www.autodesk.com/maya>, 10. 6. 2012
http://en.wikipedia.org/wiki/3D_computer_graphics_software, 10. 6. 2012
http://en.wikipedia.org/wiki/Comparison_of_3D_computer_graphics_software, 10. 6. 2012
<http://en.wikipedia.org/wiki/Photoshop>, 10. 6. 2012
http://en.wikipedia.org/wiki/Comparison_of_raster_graphics_editors, 10. 6. 2012

http://en.wikipedia.org/wiki/Image_editing, 10. 6. 2012
http://en.wikipedia.org/wiki/Graphic_art_software, 10. 6. 2012
<http://en.wikipedia.org/wiki/Paint.NET>, 10. 6. 2012
<http://intopii.com/into/>, 10. 6. 2012
<http://www.swishzone.com/index.php>, 10. 6. 2012
http://en.wikipedia.org/wiki/Comparison_of_vector_graphics_editors, 10. 6. 2012
<http://de.wikibooks.org/wiki/Inkscape>, 10. 6. 2012



Vprašanja za preverjanje znanja

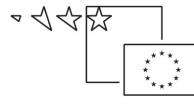
1. Kaj je računalniška grafika?
2. Kako delimo vrste grafik?
3. Kaj je digitalna slika?
4. Kako nastane digitalna slika?
5. Kako delimo digitalne slike (grafiko)?
6. Kaj je rastrska slika?
7. Kaj je vektorska grafika?
8. Kaj se zgodi pri povečanju majhne poševne rastrske premice?
9. Kaj se zgodi pri povečanju majhne vodoravne ali navpične rastrske premice?
10. Kako so zgrajene pisave na računalniku in kako se delijo pisave po končnici?
11. Kdo in kdaj je prvi uporabil matematični zapis za vektorsko Bézier krivuljo?
12. Kaj so krivulje Bézier?
13. Kakšen program je Inkscape in za kaj je namenjen?
14. Opiši vektorski format SVG.
15. Naštej štiri glavne (pomembne) lastnosti formata SVG.
16. Kaj so NURBS-krivulje in kje se te uporabljajo?
17. Opiši postopek upodabljanja risb SVG v spletnem brskalniku?
18. Kaj so pogoni spletnih brskalnikov za upodabljanje?
19. Naštej imena vsaj 6 programov za urejanje vektorske in rastrske grafike in glavno področje uporabe posameznega programa.

Odgovori

4. 1 Beseda grafika ima grški izvor v besedi γραφικός (graphikos) in je vizualna predstavitev (na neki površini, kot je stena, platno, zaslon, papir ali kamen, ki obvešča, prikazuje ali je namenjena zabavi). Sicer sama beseda grafika izhaja iz besede graf.

4. 2 Ločimo več vrst grafik, in sicer:

- reproduktivna ali [industrijska grafika](#)



- [umetniška ali originalna grafika](#)
- [računalniška grafika](#)

4. 3 Digitalna slika lahko nastane z upodabljanjem (računanjem vrednosti) ali z digitalno kamero in je dvodimenzionalni prikaz ali zapis slike s pomočjo končne množice digitalnih vrednosti najmanjših gradnikov, ki so pri rastrskih slikah slikovne točke in pri vektorskih slikah matematične formule (parametrični opis formula ali krivulj). Skrajšano povedano je dvodimenzionalni prikaz ali zapis slike s pomočjo končne množice digitalnih vrednosti, ki se imenujejo slikovne točke ali matematične formule (parametrični opis).
4. 4 S pomočjo digitalne kamere (fotoaparata), optičnega bralnika (skener), zračnega radarja, seizmološkimi profiliranjema, z računalniškim risanjem in konstruiranjem ...
4. 5 Digitalne slike (grafiko) delimo na rastrske in vektorske.
4. 6 Je sestavljena iz slikovnih točk (tipično se danes pridobi z digitalno kamero).
4. 7 Vektorska grafika je sestavljena iz matematičnih izrazov in/ali krivulj. Primerna je za tehnične skice, risbe in modeliranje ter 2D- in 3D-animacije.
4. 8 Na zaslonu se vidijo t. i. alias oz. »stopnice«. Na rastrskih slikah, ki imajo na sliki poševne črte, se po skaliranju pojavljajo nazobčani robovi oz. »stopnice«. Da se tem vidnim pojavom izognemo, se uporabi antialias oz. sosednje slikovne točke se pojavljajo deloma s to in sosednjo barvo (s tem »stopničenjem« ni več vidno).
4. 9 Premica se poveča (ne vidijo se stopnice), le robovi so bolj zabrisani.
4. 10 Se lahko skoraj poljubno poveča, ker je sestavljena iz matematičnih izrazov ali parametrov. Vsaka pisava oz. vsaka črka (ali drugi znak) oz. glif je prava risba, ki je nastala pod avtorskim peresom in pozneje shranjena v datoteko s pisavo. Te imajo večinoma končnico OTF, TTF ali PS1
4. 11 Francoski matematik Pierre Bézier leta 1962, pri oblikovanju avtomobilskih karoserij.
4. 12 Bézier krivulje so parametrične krivulje, pomembne v računalniški grafiki, za ustvarjanje vektorskih slik. Krivuljo določajo štiri točke: začetni in končni položaj ter dve ločeni vmesni (kontrolnih) točki. Bézier-jeve predmete lahko spreminjamo s premikanjem teh točk ali ročic s pomočjo miške. Matematično so te krivulje izražene s polinomi prve, druge in tretje stopnje:

$$B(t)=(1-t)P_0, t \in [0,1]$$

$$B(t)=(1-t)^2P_0+2t(1-t)P_1+t^2P_2, t \in [0,1]$$

$$B(t)=(1-t)^3P_0+3t(1-t)^2P_1+3t^2(1-t)P_2+t^3P_3, t \in [0,1],$$
kjer je parameter t del uniformnega prostora in se lahko giblje v intervalu $[0, \dots, 1]$ in s črkami P_0, P_1, P_2, P_3 so poimenovane kontrolne točke.
4. 13 Inkscape je zelo popularen odprtokodni program za risanje vektorskih risb (<http://inkscape.org/>), ki v osnovi shranjuje v formatu SVG, ki je bil razvit in normiran s strani W3-konzorcija. Ta format se uporablja za vse vrste skic in risb na spletu, npr. v največji odprti spletni enciklopediji Wikipediji.
- 4. 14 Scalable Vector Graphics** ([kratica SVG](#), [slovensko](#) skalirna *vektorska grafika*) je [označevalni jezik](#) za opis dvorazsežne statične ali animirane in risane [vektorske grafike](#) v [XML](#).
4. 15 SVG je osnovana na jeziku XML. Ta format se lahko v vsakem trenutku DTD validira. Kar je še najbolj pomembno, to je popolnoma besedilni format zapisa. S pomočjo dokumentnega objektnega modela (Document Object Model (DOM)) je mogoč direkten dinamičen dostop direktno do vsakega opisa znotraj SVG-datoteke.
- XLink-podpora: z XLink (XML Linking Language) nadbesedilnih povezav (Hyperlinks) je možno ustvariti SVG-grafike, ki so občutljive na povezave (slike s povezavami). Povezave vodijo do lokalnih ali globalnih ciljev, ki so kjerkoli v svetu.
 - Zoomiranje (skaliranje) - in možnost zasuka: vektorske grafike je mogoče skalirati brez izgub.

•SMIL-vmesnik: s SMIL (angl. Synchronized Multimedia Integration Language - sinhronizacijski večpredstavnostni integracijski jezik) je mogoče izdelati dinamične SVG-grafike z drugimi elementi, kot je npr. zvok.

4. 16 NURBS-krivulje »Non Uniform Rational Basis Spline«, kar se prevaja kot neuniformne racionalne B-krivulje. To so matematične krivulje in ploskve, ki se uporabljajo v področju računalniške grafike, na primer v CGI (angl. Computer Generated Imagery), CAD (angl. Computer Aided Design) ali za modeliranje poljubnih oblik. Predstavitev geometrijskih podatkov se izvede preko nesistematično funkcionalno določene geometrije elementov. Kontrolne točke niso nujno na sami krivulji oz. ploskvi, temveč so praviloma izven njih in so »težnostno silo« povezane s ploskvijo (kot z nekakšnimi nevidnimi elastičnimi trakovi).

4. 17 Vektorska grafika (preden se v spletnem brskalniku vidi, je potrebno enako kot za spletne strani, da se koda razčleni (izvede t.i. angl. Parsing) in potem upodobi oz. pretvori v rastrsko obliko, ki se prikaže na zaslonu).

Pri rasterizaciji, ki je lahko strojna ali programska (enako se naredi pred tiskanjem dokumenta na tiskalniku), se najprej po konturah prepozna slika in glede na gostoto rastra določi, katera slikovna točka se uporabi oz. izriše za zunanji obris objekta npr. hiše. Na koncu se z barvo pobarva notranjost hiše skladno z originalnim predlogom.

4. 18 Spletni brskalnik pred upodabljanjem slike na zaslonu vso kodo, ki jo dobi od spletnega brskalnika razčleni (angl. Parsing) in na osnovi algoritma za upodabljanje le-tega kodo prikazuje bolj ali manj skladno s standardom. Za upodabljanje slike na zaslonu spletni brskalniki uporabljajo pogon za upodabljanje t.i. angl. Rendering-Engine.

4. 19 GIMP (**G**NU **I**mage **M**anipulation **P**rogram) (za obdelavo fotografije, risanje, izdelavo poljubne fotomontaže, 2D-animacije, fraktale ...), Inkscape (je odprtokodni program, namenjen za vektorsko risanje v formatu SVG), Blender (je odprtokodni program), Adobe Photoshop (plačljiv program, podoben GIMP-u), Adobe Illustrator (plačljiv program za vektorsko risanje), Adobe Flash (komercialen program za 2D-animacijo), Autodesk Maya (komercialen program za 3D-animacijo) in Synfig Studio (je odprtokodni program za 2D-animacijo).

Učna situacija 4: O bitih in bajtih

3D in drugi digitalni podatki se v računalnikih obdelujejo ter zapišejo na pomnilniški medij v bitih. Beseda **bit** je kratica iz dveh angleških besed 'binary digit', kar pomeni dvojiško ali binarno število.

Po drugi strani pa je bit enota za merjenje količine informacije. 1 bit je najmanjša količina informacije, ki jo dobimo z odgovorom na vprašanje z dvema enako verjetnima možnostma 1 ali 0: npr. žarnica sveti ali ne, dan in noč ... V primeru, da število izidov ni le dva temveč več, je potrebno vzeti več bitov (tabela 4).

1 bajt = 1 zlog = 8 bitov³ ali s kraticami 1 B = 8 b

1. Primer enega bajta (vrednost je izmišljena): 1 B = 10001000 (2) = 136 (10).

2. Primer enega bajta (vrednost je izmišljena): 1 B = 10000010 (2) = 130 (10).

Seveda je lahko namesto podanega zaporedja osmih 1 ali 0 poljubna kombinacija vključno z osem enk ali ničel.

Tabela 4: Večkratnik *bit*-a med decimalno in dvojiško vrednostjo predpone

SI decimalne predpone		Binarna raba	IEC binarna predpona	
Ime (Simbol)	Vrednost		Ime (Simbol)	Vrednost
kilobit (kbit)	10 ³	2 ¹⁰	kibibit (Kibit)	2 ¹⁰
megabit (Mbit)	10 ⁶	2 ²⁰	mebibit (Mibit)	2 ²⁰
gigabit (Gbit)	10 ⁹	2 ³⁰	gibibit (Gibit)	2 ³⁰
terabit (Tbit)	10 ¹²	2 ⁴⁰	tebibit (Tibit)	2 ⁴⁰
petabit (Pbit)	10 ¹⁵	2 ⁵⁰	pebibit (Pibit)	2 ⁵⁰
exabit (Ebit)	10 ¹⁸	2 ⁶⁰	exbibit (Eibit)	2 ⁶⁰
zettabit (Zbit)	10 ²¹	2 ⁷⁰	zebibit (Zibit)	2 ⁷⁰
yottabit (Ybit)	10 ²⁴	2 ⁸⁰	yobibit (Yibit)	2 ⁸⁰

Iz dvojiških števil lahko pretvorimo binarni zapis v nam bolj znano desetiško število. Nanizanih je nekaj primerov pretvorbe dvojiške v desetiške vrednosti:

$$11111111 (2) = 255 (10)$$

$$11111110 (2) = 254 (10)$$

$$11111101 (2) = 253 (10)$$

$$11111100 (2) = 252 (10)$$

$$11111011 (2) = 251 (10)$$

$$11111010 (2) = 250 (10)$$

3 Po priporočilu standarda IEEE 1541, ki je bil sprejet leta 2008, ni nujno, da je skupina 8 bitov, so pa to povezani podatki (za razliko od okteta, ki so neuteženi).

$$\dots$$
$$10000000_2 = 128_{10}$$

$$\dots$$
$$00000010_2 = 2_{10}$$

$$00000001_2 = 1_{10}$$

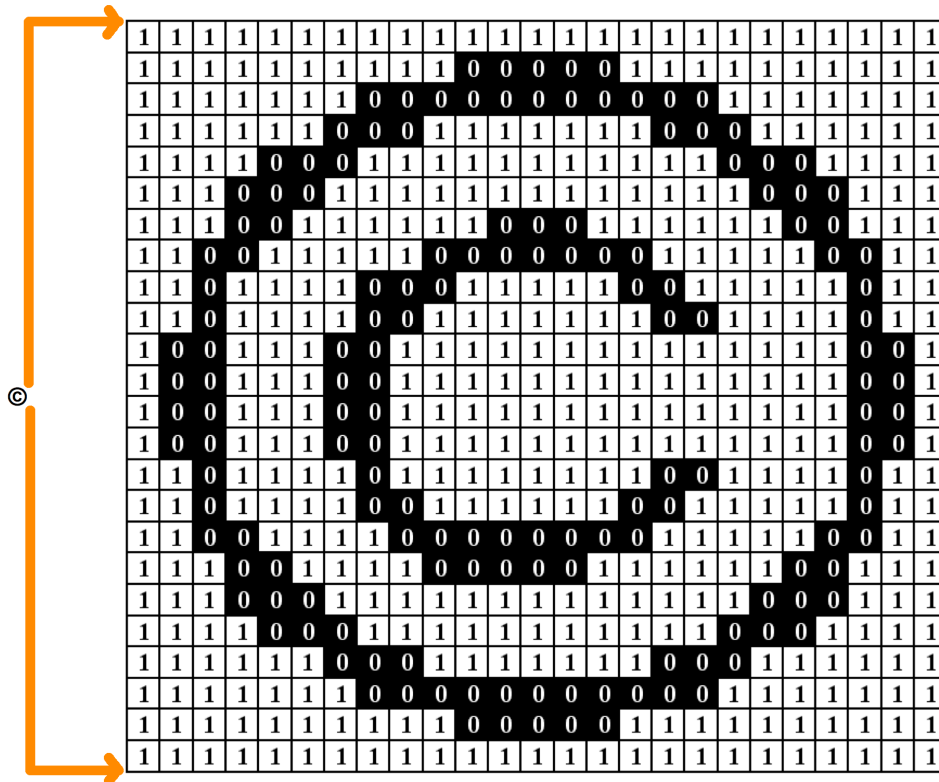
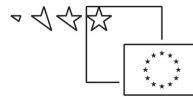
$$00000000_2 = 0_{10}$$

Po priporočilih IEEE 1541 naj se uporablja sklop enot za količine v digitalni elektroniki in računalništvu:

- bit (simbol "b"), binarna ali dvojiška številka;
- bajt angl. *byte* (simbol "B"), niz sosednjih bitov (običajno 8, vendar to ni nujno), ki so povezani v skupino;
- oktet angl. *octet* (simbol "o") je skupina osmih bitov, ki ni utežena (torej nima mestnih vrednosti) npr. 10110011_2 .

Slike, shranjene v digitalni kameri ali na računalniku, so prav tako zapisane z biti oz. bajti. Od števila bitov je odvisna velikost slikovne datoteke in s tem tudi hitrost nalaganja na računalnik ali odpiranje slik oz. video posnetkov ter animacij.

Prvi rastrski računalniški zasloni so prikazovali le črno-belo sliko (slika 104). Za posamezno slikovno točko je potrebno »prižgati« slikovno točko, če so želeli, da je ta videti belo in če se ni »prižgala«, je bila videti črno. Pri tej sliki bi ugotovili, da gre le za 1-bitno barvno globino oz. z 1 je označeno »prižgana« slikovna točka in z 0, da ni »prižgana«. Na desni strani naslednje slike so vidne vpisane binarne vrednosti in barvno pri vrednosti 1 aktivne oz. "vključene" slikovne točke (beli kvadrati), pri 0 pa niso (so črni kvadrati). Na levi strani iste slike je vidna približno originalna velikost znaka (pri 12 pt) za zaščito kopiranja oz. avtorskih pravic (ang. Copyright). Na desni je ta povečan za faktor 3600.



Slika 104: Prikaz črke za pridržane avtorske pravice na črno-belem zaslonu

Velikosti digitalnih podatkov in slikovne datoteke

Bajt (tudi zlog, byte, oktet (ni utežen)) je v računalništvu enota za merjenje količine podatkov. V računalništvu je tipično prevzeto, da je 1 bajt = 8 bitov ali s kraticami 1 B = 8 b.

Pri navajanju velikosti pomnilnika ali shranjevanju podatkov je pogosta raba predpon, osnovanih na potencah števila 2, in ne 10, torej:

$$1 \text{ K} = 2^{10} = 1.024$$

$$1 \text{ M} = 2^{20} = 1.048.576$$

$$1 \text{ G} = 2^{30} = 1.073.741.824$$

$$1 \text{ Tera} = 1 \text{ T} = 2^{40} = 1.099.511.627.776.$$

$$1 \text{ Peta (P)} = 1024 \text{ T} = 2^{50} \text{ B} = 1125899906842624$$

$$1 \text{ Eksa (E)} = 1024 \text{ P} = 2^{60} \text{ B} = 1152921504606846976$$

Preračunajmo, koliko bitov ima 1 KB: 1 KB = 1.024 B = 1.024 • 8 b = 8192 b.

Novjša oznaka je: 1 kibibit = 1 kubit = 2¹⁰ bitov = 1024 bitov.

Nekaj primerov tipičnih velikosti digitalnih podatkov glede na velikosti zapisa:

- 90 bajtov (angl. bytes): dovolj za shranjevanje tipične vrstice besedila iz knjige.
- 512 bajtov = 90 bytes = ½ KiB: tipična velikost sektorja s trdega diska.
- 1024 bytes = 1 KiB: klasična velikost bloka v UNIX-ovem datotečnem sistemu.
- 2048 bytes KiB = 2: sektor CD-ROM.
- 4096 bytes = 4 KiB: spomin na stran x86 (Intel od 80386).

- 4 kB: približno eno stran besedila iz romana.
- 120 kB: besedilo tipične žepne knjige.

Tabela 5: Večkratnik bajta (angl. *Bytes*) med decimalno in dvojiško vrednostjo

SI decimalne predpone		Binarna raba	IEC binarna predpona	
Ime (Simbol)	Vrednost		Ime (Simbol)	Vrednost
kilobyte (kB)	10^3	2^{10}	kibibyte (KiB)	2^{10}
megabyte (MB)	10^6	2^{20}	mebibyte (MiB)	2^{20}
gigabyte (GB)	10^9	2^{30}	gibibyte (GiB)	2^{30}
terabyte (TB)	10^{12}	2^{40}	tebibyte (TiB)	2^{40}
petabyte (PB)	10^{15}	2^{50}	pebibyte (PiB)	2^{50}
exabyte (EB)	10^{18}	2^{60}	exbibyte (EiB)	2^{60}
zettabyte (ZB)	10^{21}	2^{70}	zebibyte (ZiB)	2^{70}
yottabyte (YB)	10^{24}	2^{80}	yobibyte (YiB)	2^{80}

Za prenos in procesiranja digitalnih podatkov se uporabljajo predpone:

1 kilobit = 1 Kbit = 10^3 bitov = 1000 bitov

1 mebibyte = 1 MiB = 2^{20} B = 1.048.576 B

1 megabyte = 1 MB = 10^6 B = 1.000.000 B

1 gibibyte = 1 GiB = 2^{30} B = 1.073.741.824 B

1 gigabyte = 1 GB = 10^9 B = 1.000.000.000 B

Za binarne večkratnike desetiškega eksponenta se pri znaku doda črka i (tabela 6).

Tabela 6: Binarni večkratniki števila 2

Mnogokratnik	Predpona	Znak/simbol	Opis	Vrednost
2^{10}	kibi	Ki	Kilobinary	1024
2^{20}	mebi	Mi	Megabinary	1048576
2^{30}	gibi	Gi	Gigabinary	1073741824
2^{40}	tebi	Ti	Terabinary	1099511627776
2^{50}	pebi	Pi	Petabinary	1125899906842624
2^{60}	exbi	Ei	Exabinary	1152921504606846976

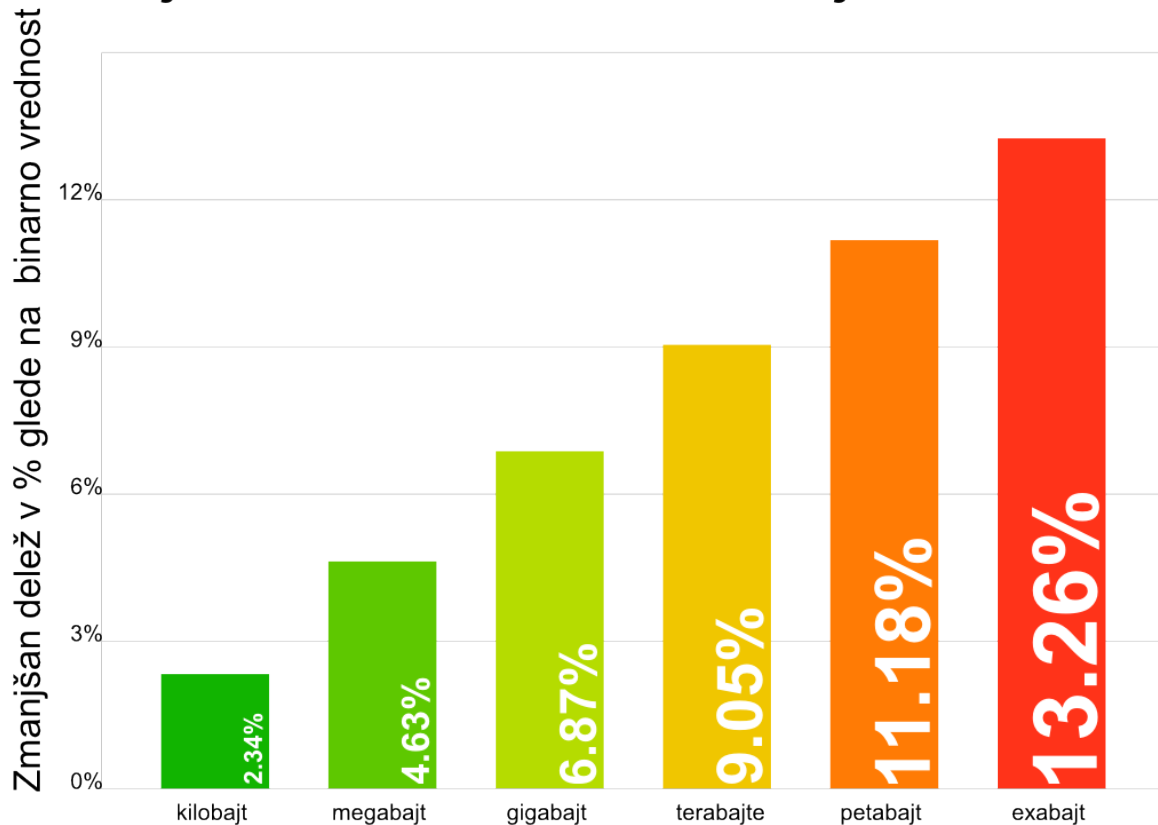
Pri preračunavanju npr. velikosti diska proizvajalca so bili navedeni podatki 1TB in ko so preračunani na TB, je to znašalo 9 % manj (razlika med faktorjem 1000 in 1024) (slika 105).

Drugi primer je, da proizvajalec diska navede podatek o velikost diska 137 GB in pri tem uporabi 81

Učno gradivo je nastalo v okviru projekta Munus 2. Njegovo izdajo je omogočilo sofinanciranje Evropskega socialnega sklada Evropske unije in Ministrstva za šolstvo in šport.

decimalni mnogokratnik: $137 \cdot 10^9$ B. Če uporabimo dvojiški mnogokratnik, je dejanska velikost diska 128 GB: $(137000000000 / 2^{30} = 128)$.

Primerjava med desetiški in dvojiškimi enotami



Metrične kapacitete shranjevanja (log. merilo)

Slika 105: Primerjava napake pri preračunavanju dvojiških in desetiških enot

Najpogosteje se kot pomnilniki uporabljajo: RAM (začasni pomnilnik z naključnim dostopom ali angl. Random access memory), trdi diski (za shranjevanje programov in podatkov na računalniku oz. strežniku), optični diski (CD (angl. *Compact Disc*, kompaktni disk), DVD (angl. *Digital Versatile Disc*) in BD (angl. Blu-ray disc) in USB-ključki (kratica USB je iz angleških besed Universal Serial Bus, kar pomeni univerzalno serijsko vodilo). Za sedanji CD je kapaciteta 700 MB (redkeje 650 ali 800 MB), enoslojni DVD 4,7 GB, enostranski BD je 25 GB in dvostranski BD je 50 GB, polprevodniški disk SSD (angl. Solid-State Drive) s tipično velikostjo 256 GB ali 512 GB (za prenosne naprave Apple so to tudi 16, 32, 64 GB).

Več o predponah, binarnih podatkih in diskih SSD najdete je na spletnih naslovih:

http://en.wikipedia.org/wiki/IEEE_1541, 16. 7. 2012,

http://en.wikipedia.org/wiki/Solid-state_drive, 16. 7. 2012.

http://en.wikipedia.org/wiki/List_of_solid-state_drive_manufacturers, 16. 7. 2012.

Barvni obseg in globina

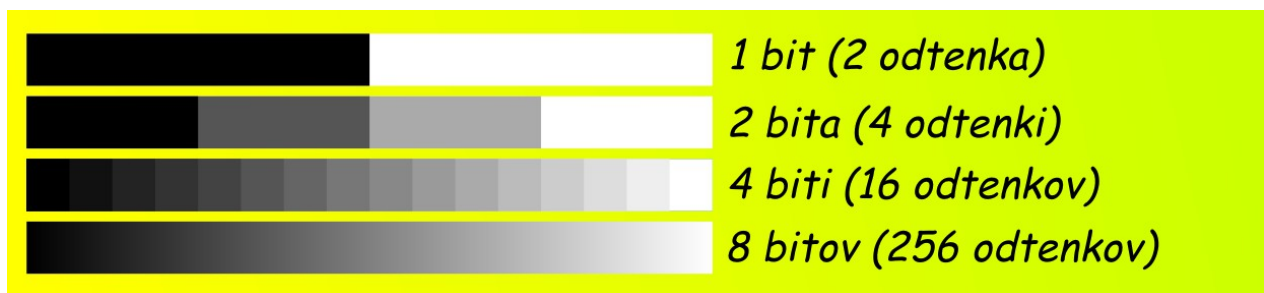
Barvni obseg določa število vseh barv (angl. color gamut), ki jih lahko prikažemo na zaslonu ali

npr. tiskamo na tiskalniku (slika 106 in 107).

Število odtenkov, ki jih dobimo iz razpoložljivega števila bitov, izračunamo na osnovi enačbe:
 $2^{\text{št. bitov}} = \text{št. barvnih ali sivinskih odtenkov}$ (4.4.1).

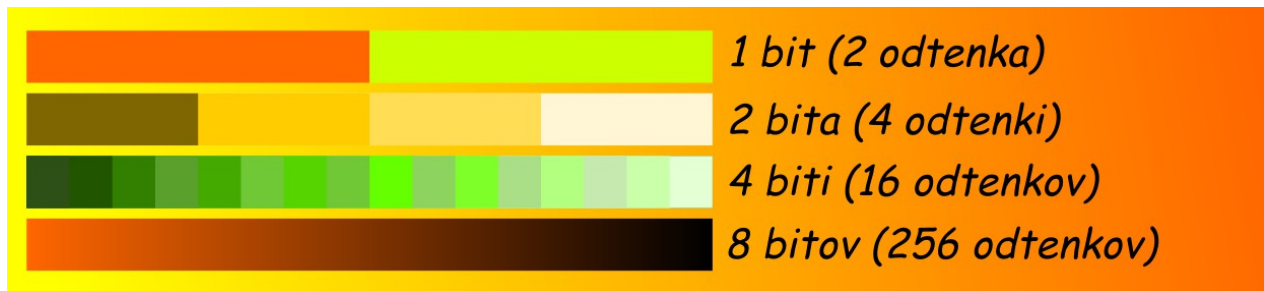
Določa število bitov, ki se uporabljajo za zapis barv na rastrski sliki. Običajno so uporabljaja 1 bit, 8 bitov, 24 bitov ali 32 bitov.

1. Primer: z 1 bitom se dobi $2^1 = 2$ odtenka (črno-bela slika).
2. Primer: s 4 biti se dobi $2^4 = 16$ odtenkov.
3. Primer: s 7 biti se dobi $2^7 = 128$ odtenkov.
4. Primer: s 16 biti se dobi $2^{16} = 65536$ odtenkov.
5. Primer: s 24 biti se dobi $2^{24} = 16\,777\,216$ odtenkov.



Slika 106: Primer sivinskih odtenkov glede na število bitov, potrebnih za zapis le-teh

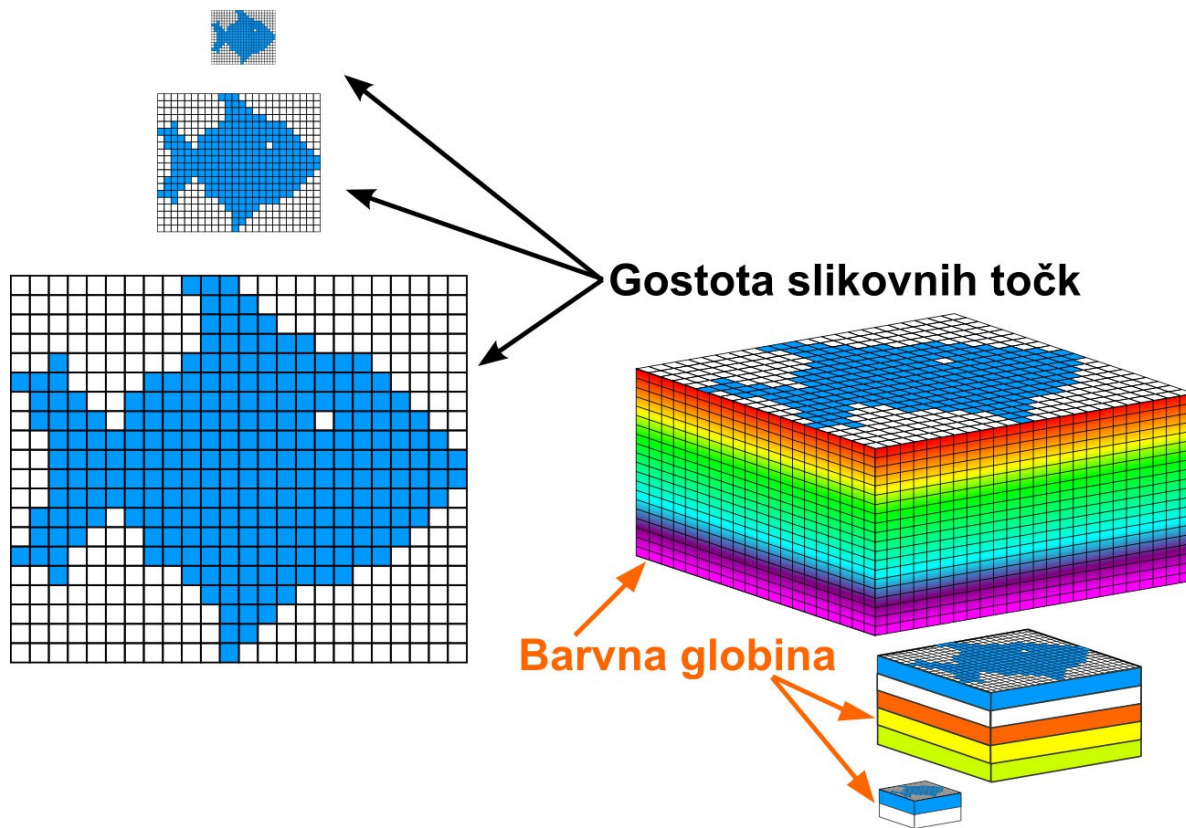
Zaradi računalniškega načina zapisa podatkov po 8 bitov, ki tvorijo 1 zlog ali byte, se za barvno globino najpogosteje uporabljajo le večkratniki 8 bitov.



Slika 107: Primer barvnih odtenkov glede na število bitov, potrebnih za zapis le-teh

Barvna globina je pojem, ki pove, koliko barv lahko zavzema neka slikovna točka. Na levem delu slike 108 je ponazorjena velikost oz. gostota slikovnih točk (manjše kot so slikovne točke, manj se razločijo). Na desnem delu slike 108 je ponazorjena barvna globina, za najmanjšo sliko bi bila dovolj 1-bitna globina, saj z njim lahko zapišemo dve kombinaciji (belo in modro barvo). Srednji »sedvič« s sliko ribe ima le barvno globino petih barv (katero želimo, bi lahko sami izbrali in bi z njo npr. pobarvali okolico ali oko ribe). Sicer je slika na navadnem računalniškem zaslonu ploska in nima debeline. Z desno sliko 108 je samo ponazorjeno, da so barve pod prvo plastjo le možnosti, ki se lahko uporabijo, če je to potrebno oz. smiselno.

Skupaj z gostoto slikovnih točk je barvna globina matrike slike osnova vsake rastrske grafike. Obe vrednosti skupaj določata največjo teoretično dosegljivo kakovost slike.



Slika 108: Ponazoritev gostote slikovnih točk in in simboličen prikaz barvne globine

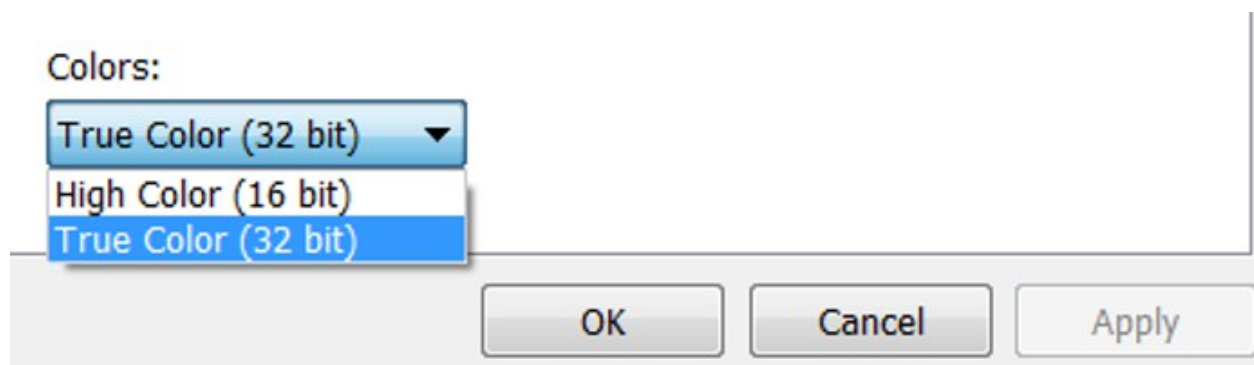
V naslednji tabeli 7 so zbrani podatki o najpogostejših barvnih globinah, imenu ali uporabi, kodiranju (število barv na kanal) in število kombinacij oz. barv, ki ga omogoča barvna globina.

Tabela 7: Najpogosteje uporabljane barvne globine

Barvna globina	Ime/Uporaba	Kodiranje	Št. barv
1-bitna	Monokromatska	Ni enoznačne prireditve	$2^1 = 2$
4-bitna	Uporaba pri EGA-grafičnih karticah	Ni enoznačne prireditve	$2^4 = 16$
6-bitna	Uporaba pri Amiga računalnikih za HAM -in način Halfbrigh	Ni enoznačne prireditve	$2^6 = 64$ (s pomočjo posebnega HAM-mehanizma vse do 4096)
8-bitna	Uporaba pri MSX2 -računalnikih	Rdeča: 3 biti Zelena: 3 biti Modra: 2 bita	$2^8 = 256$ (s pomočjo posebnega HAM8 -mehanizma pri rač. Amigi, vendar vse do okrog 2 milijonov)

Barvna globina	Ime/Uporaba	Kodiranje	Št. barv
12-bitna	Uporaba pri različnih <u>NeXT</u> -delovnih postajah	4 biti na barvo	$2^{12} = 4096$
15-bitna	<u>Real Color</u> (realna barva)	Rdeča: 5 bitov Zelena: 5 bitov Modra: 5 bitov	$2^{15} = 32.768$
16-bitna	<u>High Color</u> (visoka barva)	Rdeča: 5 bitov Zelena: 6 bitov Modra: 5 bitov	$2^{16} = 65.536$
24-bitna	<u>True Color</u> (resnična barva)	Po en bajt oz. <u>Byte</u> (8 bitov) za R, G in B	$2^{24} = 16.777.216$
24-bitne barve + 8-bitna Alpha	True Color z 8-bitnim <u>Alphakanalom</u>	En Byte (8 bit) za R, G in B in α	$2^{24} = 16.777.216$
30-bitna	npr. <u>PAL</u>	Po 10 bitov za Y, U in V	$2^{30} = 1.073.741.824$
36-bitna	npr. visokokakovostna fotografija	Po 12 bitov za R, G in B	$2^{36} = 68.719.476.736$
42-bitna	npr. visokokakovostni LCD ali plazma televiziji	Po 14 bitov za R, G in B	$2^{42} = 4.398.045.511.104$
48-bitna	npr. visokokakovostni optični bralnik	Po 16 bitov za R, G in B	$2^{48} = 281.474.976.710.656$

Pri optičnim branju (skeniranju), kinu, TV, pri tiskanju se pogosto pojavljajo tudi barvne globine s 30, 32, 36, 40 in 48 biti. Za računalniške zaslone je tipično, da imamo 32 bitov (3 x 8 za po en kanal RGB in 1 x 8 za Alfa kanal) - slika 109.



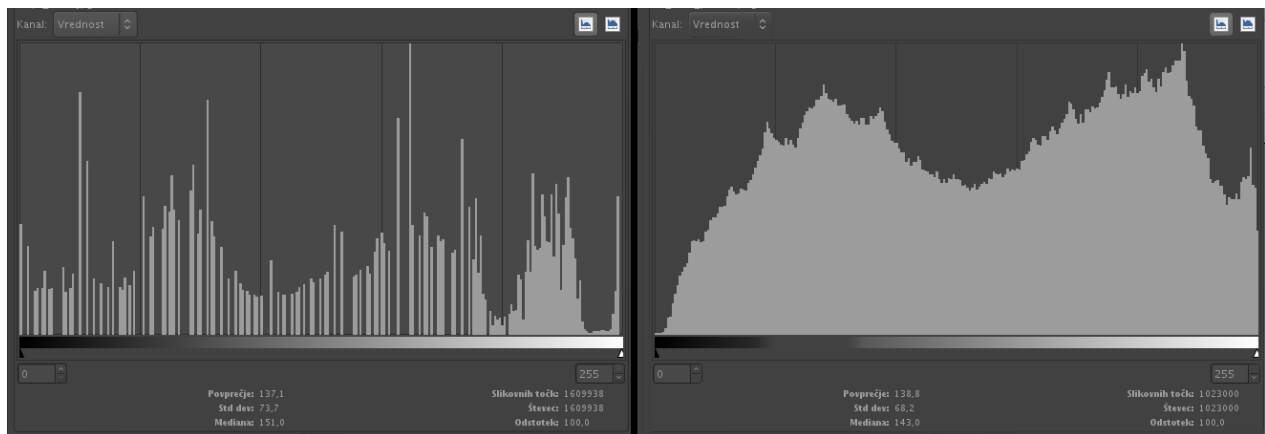
Slika 109: Sprememba barvne globine pri OS MS Windows 7

Glede na barvno globino je možno deliti slike na:

1. Barvne, ki so tonsko zvezne (ponavadi 24-, 32- ali 48-bitne).
2. Sivinske, ki so tonsko zvezne (ponavadi 8- ali 16-bitne).
3. Indeksirane barvne (sivinske) ali tonsko nezvezne (ponavadi do 256 barvnih odtenkov ali še manj oz. več).
4. Indeksirano sliko le z dvema odtenkoma imenujemo črno-belo (samo črna in bela akromatska vrednost).
5. Dvotonske slike kot kombinacijo določenih barvnih odtenkov iste barve (od rumene do črne ali rdeče do črne, sepia je rumenkasto-rjava ali druga kombinacija).
6. Tri- ali še večtonska slika, ki je kombinacija določenih barvnih odtenkov različnih barv (npr. za tritonsko sliko so to recimo odtenki rdeče, rumene in modre barve).

Tonsko zvezna slika pomeni, da ima slika različne odtenke brez vmesnih vrzeli. Zveznost tonskega obsega se lepo vidi na histogramu (slika 86).

Če so prisotne vrzeli v histogramu, se to na sliki vidi umetno kot področja, ki bi bila pobarvana npr. s čopičem.



Slika 110: Nezvezni in zvezni histogram

Več o 3D-zaslonih in stereoskopiji je na spletnih naslovih:

http://en.wikipedia.org/wiki/3D_displays, 16. 7. 2012

<http://en.wikipedia.org/wiki/Stereoscopic>, 16. 7. 2012

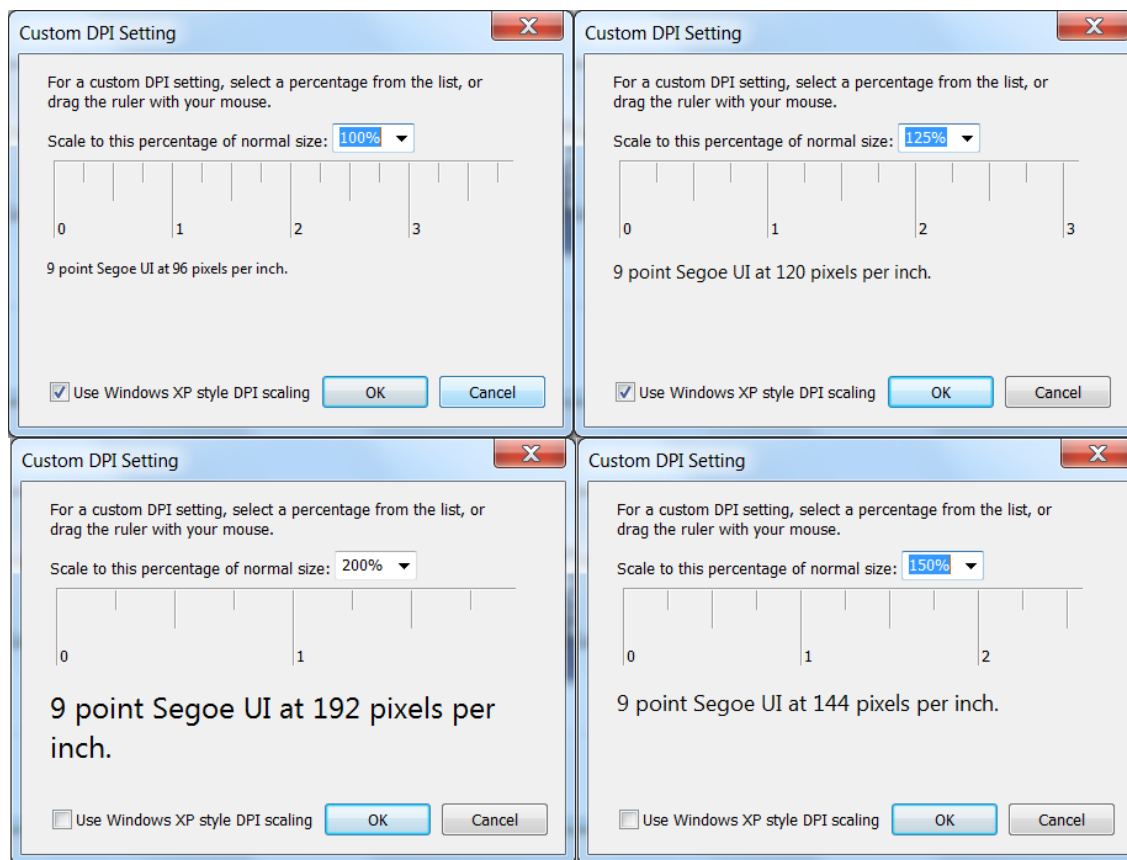
Gostota točk

Gostota točk (pik) je pri reprodukciji slike mera za raven podrobnosti za rastrski prikazovalnik in tako eden od vidikov kakovosti tehničnega načina prikazovanja. Točke gostote so prikazane na primer s 4-barvnim tiskom ali pri prikazu na zaslonu.

Običajne enote za gostoto točk so v praksi:

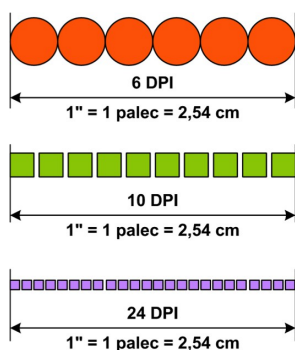
- dpi (angl. dots per inch,) ali slov. "pik na palec" (slika 111),
- ppi (angl. pixel per inch,) ali slov. "slikovnih pik na palec",

- lpi (angl. lines per inch) ali slov. "linij na palec".



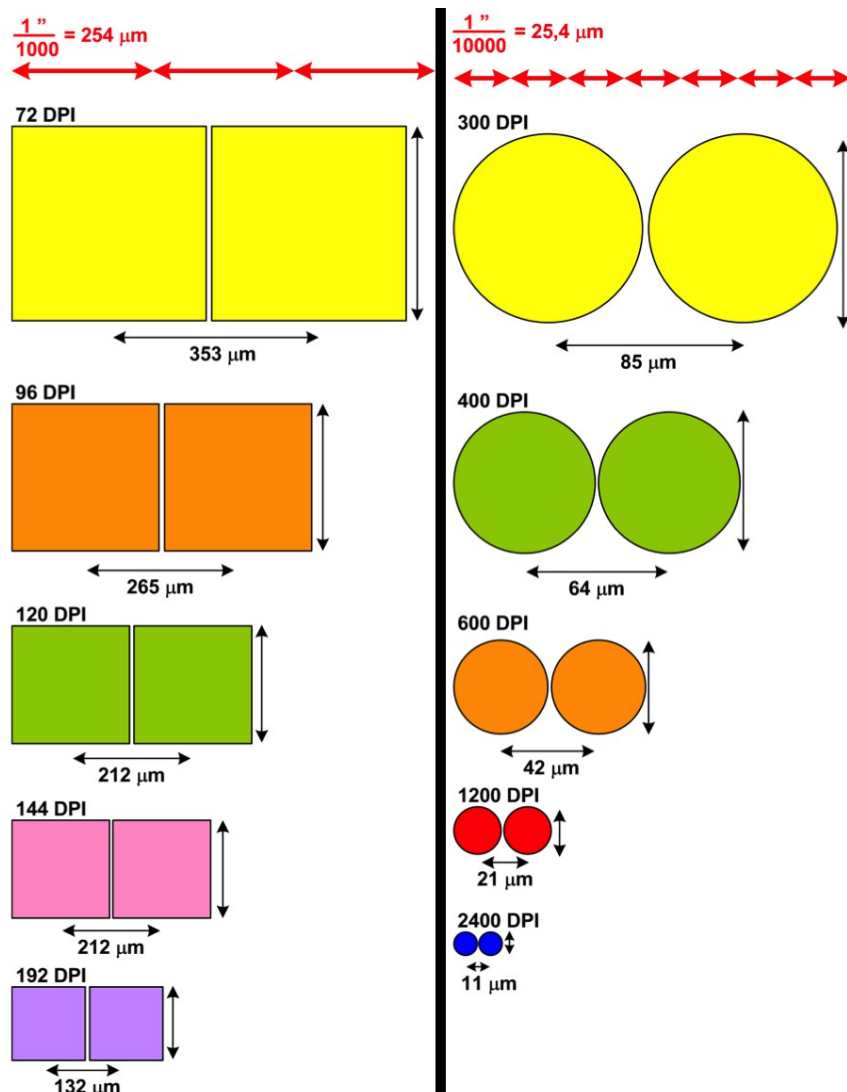
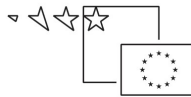
Slika 111: Sprememba DPI na zaslonu pri OS Windows 7

DPI je enota za merjenje ločljivosti pri tiskanju ali prikazu na zaslonu, ki predstavlja število natisnjenih točk (pik) na palec (slika 112).



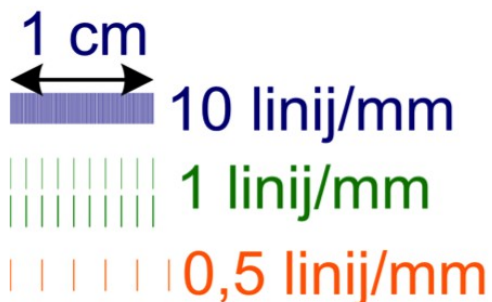
Slika 112: Ponazoritev, kako naj si poenostavljeno predstavljamo DPI

Oblika, postavitev in velikost točk pri tiskanju je silno odvisna od uporabljene tehnologije oz. tehnike tiska. Običajno je potrebna vsaj trikrat večja ločljivost začetne slike, kot je to potrebno za enako kakovosten prikaz na zaslonu (slika 113).



Slika 113: Ponazoritev idealnih velikosti točk DPI pri zaslonu (levo) in tiskanju (desno)

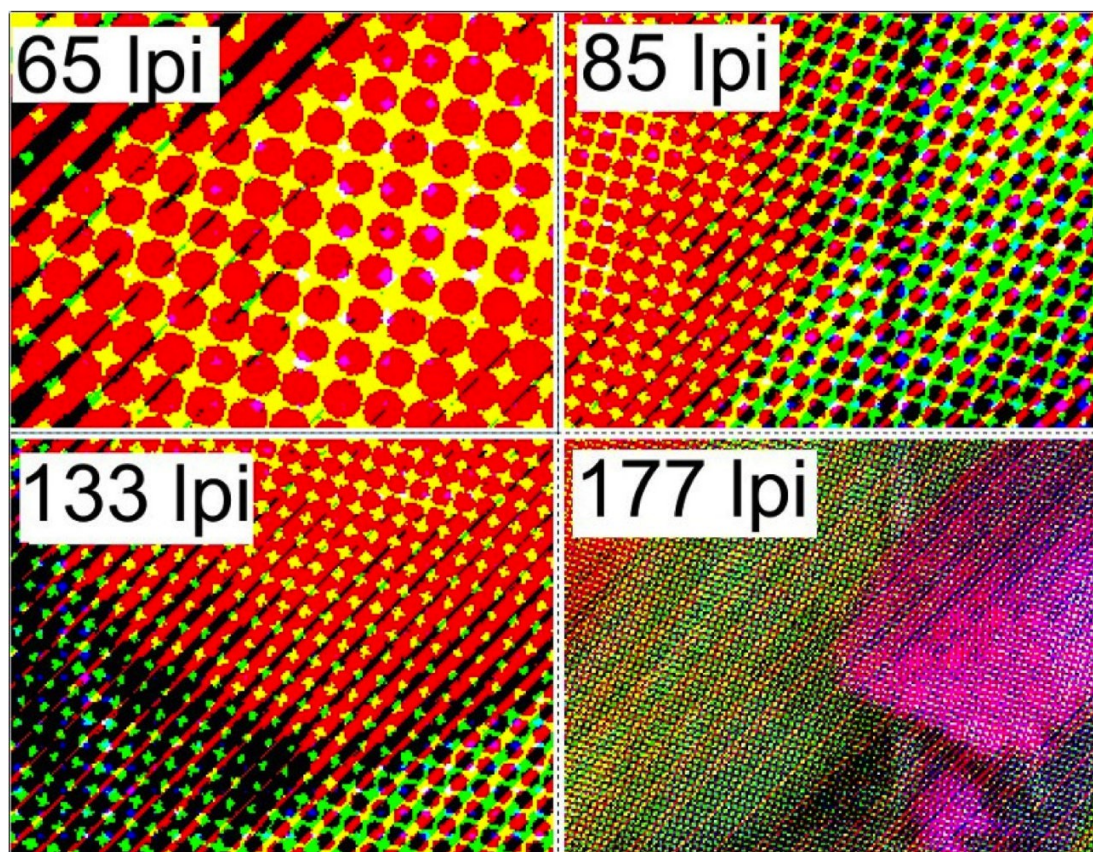
LPI predstavlja ločljivost človeškega očesa glede črt (oz. poltonskih točk - slika 114). Primer: za ustvarjanje različnih svin je slika razdeljena na posamezne (poltonske) točke, temnejši deli so sestavljeni iz večjih točk, svetlejši deli iz manjših.



Slika 114: Število linij na cm je pri nas bolj običajno kot število linij na palec

Število poltonskih točk določa frekvenco, ki jo merimo z enoto lpi. Nižja frekvenca pomeni večje točke. Časopisi npr. običajno uporabljajo 60 do 90 lpi, revije 133 ali 150 dpi, posebne tiskovine (knjige, posterji ...) pa dosežejo običajno največ 200 lpi (slika 115).

Običajno se uporablja pravilo: $DPI = 2 \cdot LPI$, npr. pri tiskanju revije s 150 LPI naj bi se fotografije optično prebrale pri 300 DPI.



Slika 115: Prikaz poltonskih točk pri različnih vrednostih lpi

S PPI ali p/i merimo ločljivost različnih naprav, kot so računalniški monitorji, optični bralniki ali tipala digitalnih kamer.

Ta enota za merjenje ločljivosti slike določa velikost natisnjene slike, npr.: sliko velikosti 1600 •

1200 točk bomo pri ločljivosti 300 ppi natisnili na velikost 5,3 • 4 palce (14 • 10,5 cm), pri ločljivosti 180 ppi pa jo lahko natisnemo na 8,9 • 6,7 palcev (23,4 • 17,6 cm).

V tabeli 8 vidimo pretvorbo posameznih enot med vrednostmi DPI in fizično velikost točke (angl. Dot).

Tabela 8: Pretvorba posameznih enot med vrednostmi DPI in fizično velikost točke (angl. Dot)

DPI (dot/in)	dpcm (dot/cm)	Pitch (µm/dot)
72	28	350
96	38	265
150	59	169
300	118	85
2540	1000	10
4000	1575	6

Gostota točk se navede tudi pri rastrskem prebiranju slik, na primer z uporabo optičnega bralnika. Tudi tukaj je eden od vidikov kakovost postopka vzorčenja.

Na človeško oko, ki "zaznava kakovost" (vizualna percepcija), poleg gostote točk odločilno vplivajo še tudi drugi dejavniki, kot sta barvna globina in ostrina vzorca (slike).

Več o tiskanju, tipografskih enotah in merah najdete na spletnih naslovih:

http://en.wikipedia.org/wiki/Metric_typographic_units, 16. 7. 2012

http://en.wikipedia.org/wiki/Pixels_per_inch, 16. 7. 2012

http://en.wikipedia.org/wiki/Dots_per_inch, 16. 7. 2012

http://en.wikipedia.org/wiki/Lines_per_inch, 16. 7. 2012

<http://en.wikipedia.org/wiki/Dpcm>, 16. 7. 2012

<http://www.iol.ie/~sob/tm/index.shtml>, 16. 7. 2012

http://en.wikipedia.org/wiki/Pixels_per_inch, 16. 7. 2012

[Samples per inch, 16. 7. 2012](#)

[Lines per inch, 16. 7. 2012](#)

[Metric typographic units, 16. 7. 2012](#)

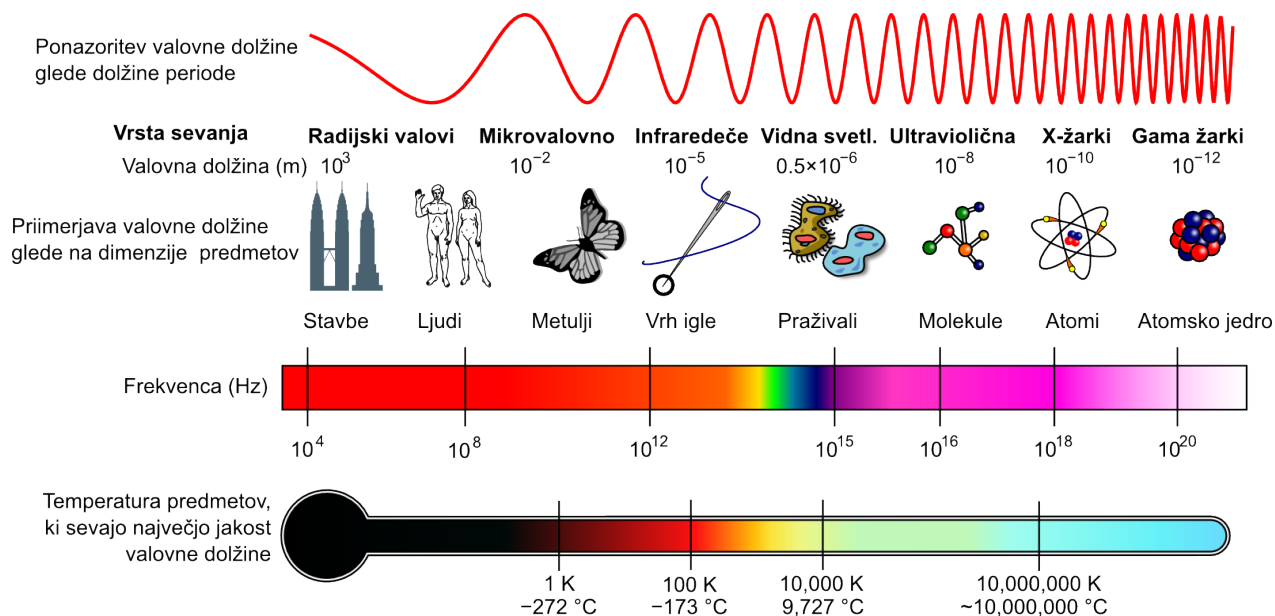
[Display resolution, 16. 7. 2012](#)

[Mouse DPI, 16. 7. 2012](#)

[Twip, 16. 7. 2012](#)

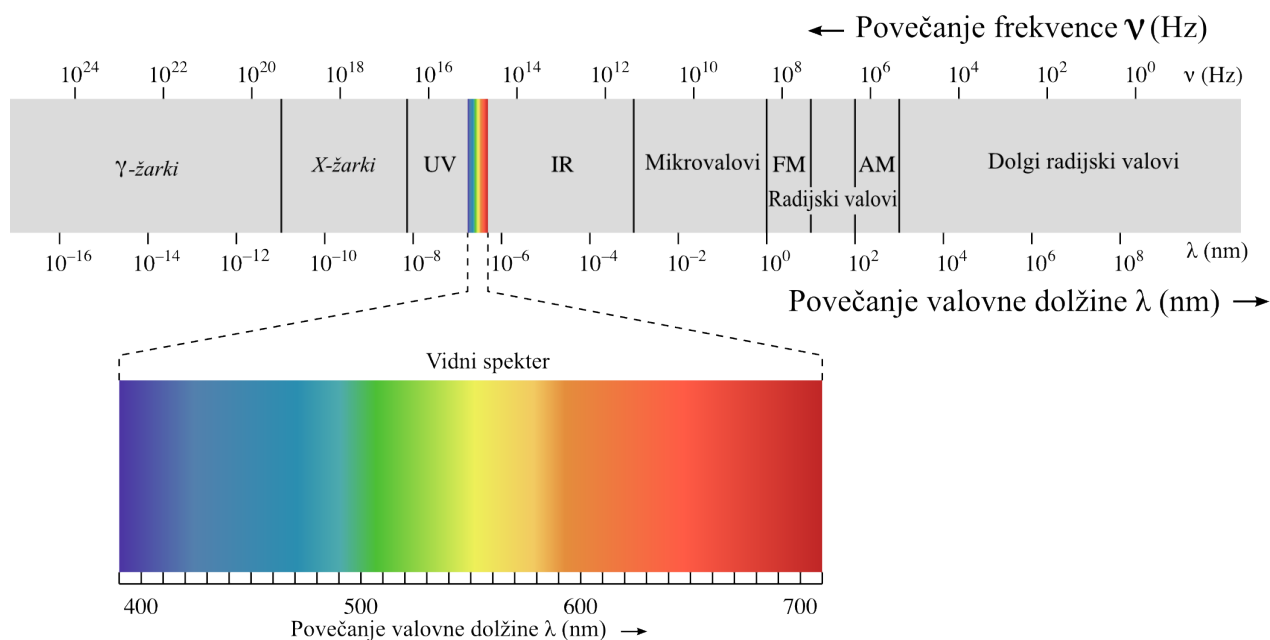
Prikaz slike na zaslonu

Za lažje razumevanje snovi si pogledjmo nekaj pojmov o ustvarjanju in prikazu barv na računalniškem zaslonu ali s pomočjo projektorja. Svetloba je del elektromagnetnega spektra, podobno kot infrardeča in ultravijolična (slika 116).



Slika 116: Prikaz elektromagnetnega spektra in sevanje črnega telesa

Barva je elektromagnetno valovanje med 470 in 780 nm (slika 117) oz. z očesom vidno zaznavanje, določeno z odtenkom, nasičenostjo in svetlostjo. To valovanje zaznamo z očmi kot lastnost stvari, da predmete vidimo.



Slika 117: Umestitev vidnega spektra v elektromagnetni spekter

Barvni model je sistem za predstavitev barv (tudi teorija) s tremi ali več osnovnimi merami. Za različne namene obstajajo različni modeli.

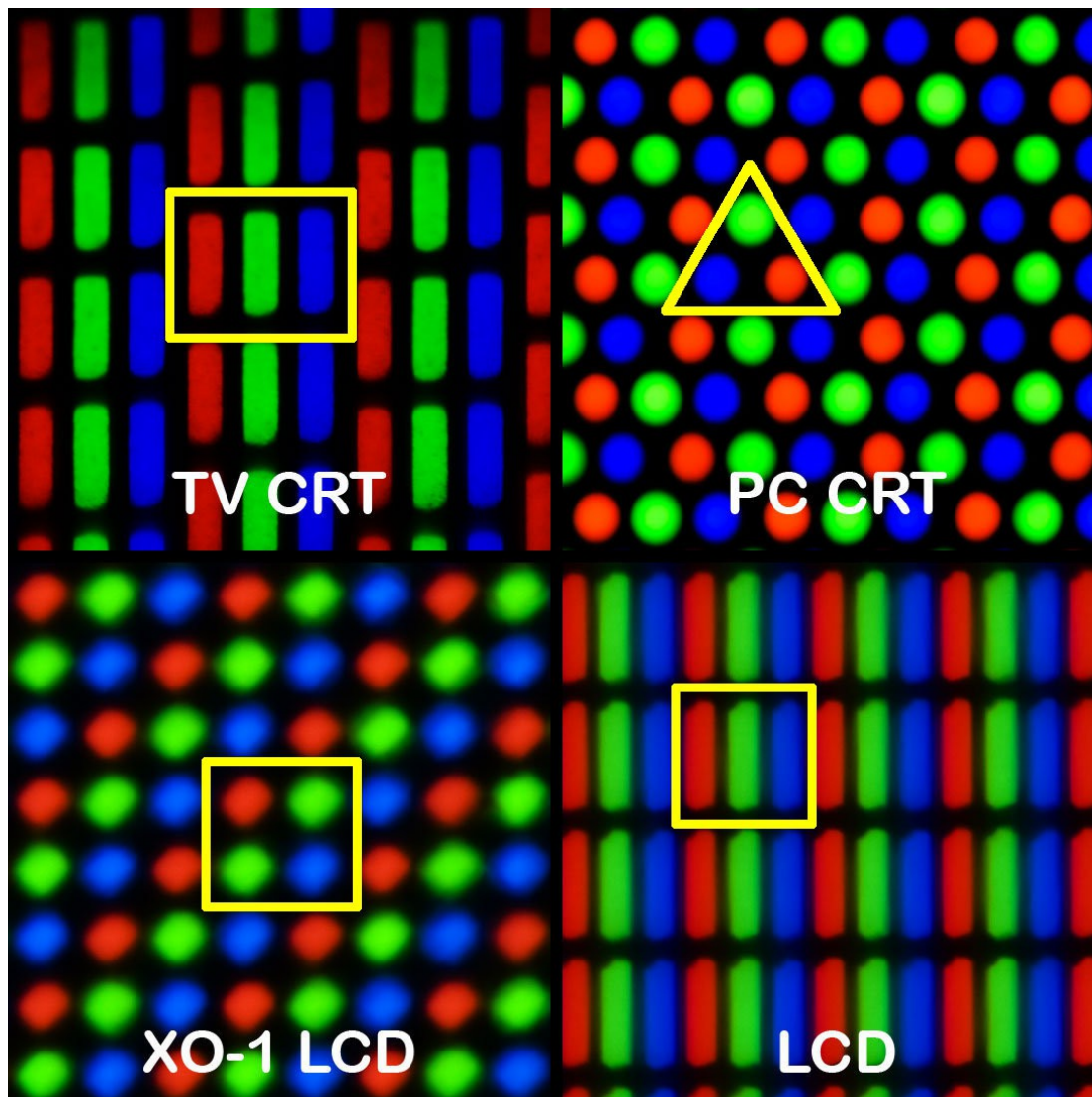
Barvna paleta je množica barvnih odtenkov, ki definirajo barvni prostor. Paleta najpogosteje vsebuje 256 barv, ki so lahko poljubno izbrane iz barvnega prostora s 16 777 216 odtenki (24 bitov).

Barvna lestvica je obseg barv, ki ga lahko reproduciramo z napravo ali sistemom.

Barvno upravljanje (angl. color management) je sistem upravljanja izhodnih podatkov vseh naprav v produkcijski verigi, ki zagotavlja, da so končni rezultati zanesljivi in ponovljivi z vidika reprodukcije barv.

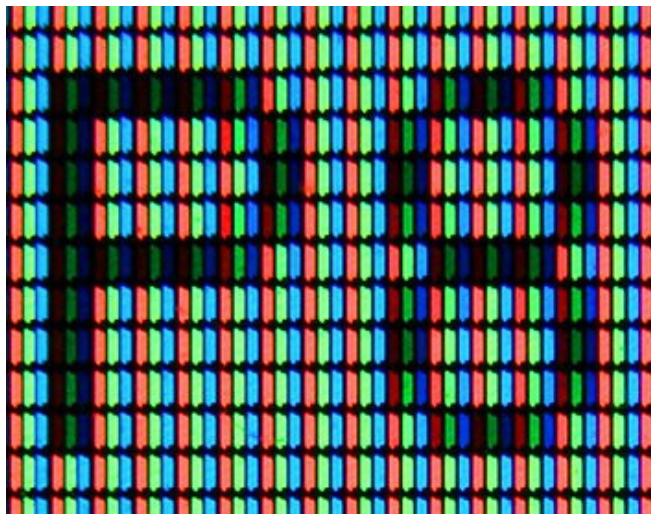
Barvni prostor je teoretični obseg barv, ki jih zmore neka naprava reproducirati ali jih pod določenimi pogoji zazna človeško oko.

Slikovna točka je najmanjši element neke slike. Glede na izhodni medij ali napravo so slikovne točke po obliki in tehnično različno izvedene (obkroženo območje na sliki 118). Velikost slikovne točke pri zaslonih za kvadratno stranico znaša danes običajno med 0,25 - 0,28 mm. Za prikazovalnike (zaslone ali monitorje, TV-sprejemnike) in projektorje se uporablja t. i. barvni model RGB. Torej za eno slikovno točko sestavlja tri slikovne podtočke (angl. Subpixel, vidne v okvirju na sliki 118) v rdeči, zeleni in modri barvi. Pri tiskanju se uporablja drug model CMYK (Cyan, Magenta, Yellow, BlacK). Človeškemu vsakdanjiku je bliže barvni model HSV ali HSL ali HSI. Danes obstaja vsaj okoli 30 barvnih modelov, od katerih jih je okrog 10 najpogosteje v uporabi. Z barvnim modelom je opisana tehnična realizacija barve na določeni napravi ali kot teoretični model z namenom uporabe le-tega pri izboljšanju videza slike ali pri izbiri barv oz. oblik.



Slika 118: Različne slikovne točke glede na rastrski prikazovalnik, vir: <http://en.wikipedia.org/wiki/Subpixel>, 20. 7. 2012.

Če želimo na beli podlagi videti črne črke, se ne krmilijo podtočke, za belo barvo vklopimo vse tri z enakim največjim deležem svetlosti posameznih podtočk (slika 119), vir: <http://en.wikipedia.org/wiki/Subpixel>, 20. 7. 2012. Ostale barvne odtenke dobimo z neenakomernim oddajanjem svetlobe iz RGB-podtočk. Več o tem preberite pri barvnem modelu RGB.



Slika 119: Od blizu slikan zaslon s prikazom oznake P8, ki je visoka 8 slikovnih točk

Barvni modeli

Zapis barv in njihove modele običajno delimo na:

- CIE-barvni standardi.
- Barvne tabele (Pantone, Trumatch, Focoltone).
- Strojno orientirani barvni modeli (RGB, CMY, YUV, YIQ).
- Uporabniško orientirani barvni modeli (HSV, HLS, HLC).

Blender uporablja kot privzeti barvni model RGB in s klikom na zavihek je možno preklopiti na barvni model HSV ali Hex (šestnajstiški zapis se veliko uporablja pri barvah na spletnih straneh).

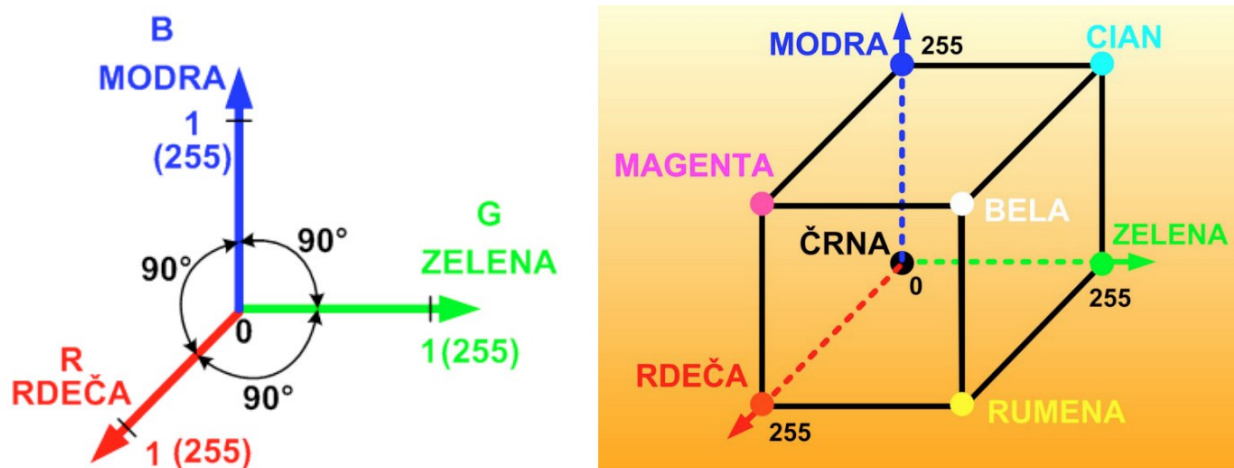


Slika 120: Na vsaki od slik je izbran eden od zavihov za spremembo barvnega modela v Blenderju 2.63

Več o barvnih modelih lahko najdete na spletnih naslovih:
http://en.wikipedia.org/wiki/Color_banding, 16. 7. 2012
http://en.wikipedia.org/wiki/Color_theory, 16. 7. 2012
http://en.wikipedia.org/wiki/List_of_palettes, 16. 7. 2012

Barvni model RGB

Barve spektra dosežemo z mešanjem treh osnovnih barv ali komponent: rdeče R (angl. Red), zelene G (angl. Green) in modre B (angl. Blue) v različnih vrednostih. Model je predstavljen v obliki kocke s pravokotnimi robovi (slika 121 in 123).

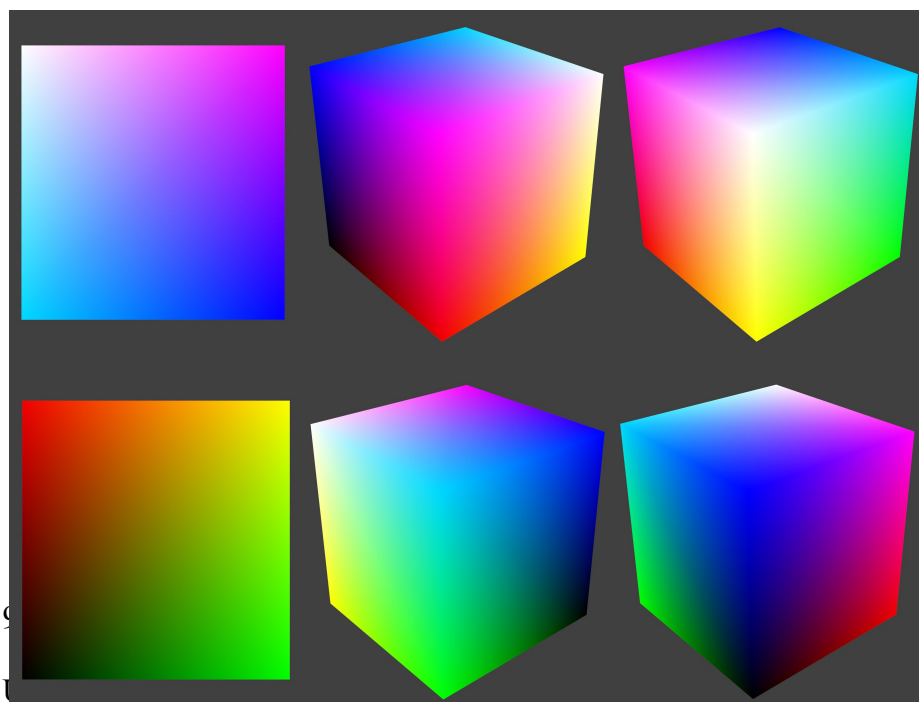


Slika 121: Trije pravokotni RGB-vektorji (levo) in RGB-kocka (desno)

Model RGB je aditivni (seštevalni model) z mešanjem barv svetlobe.

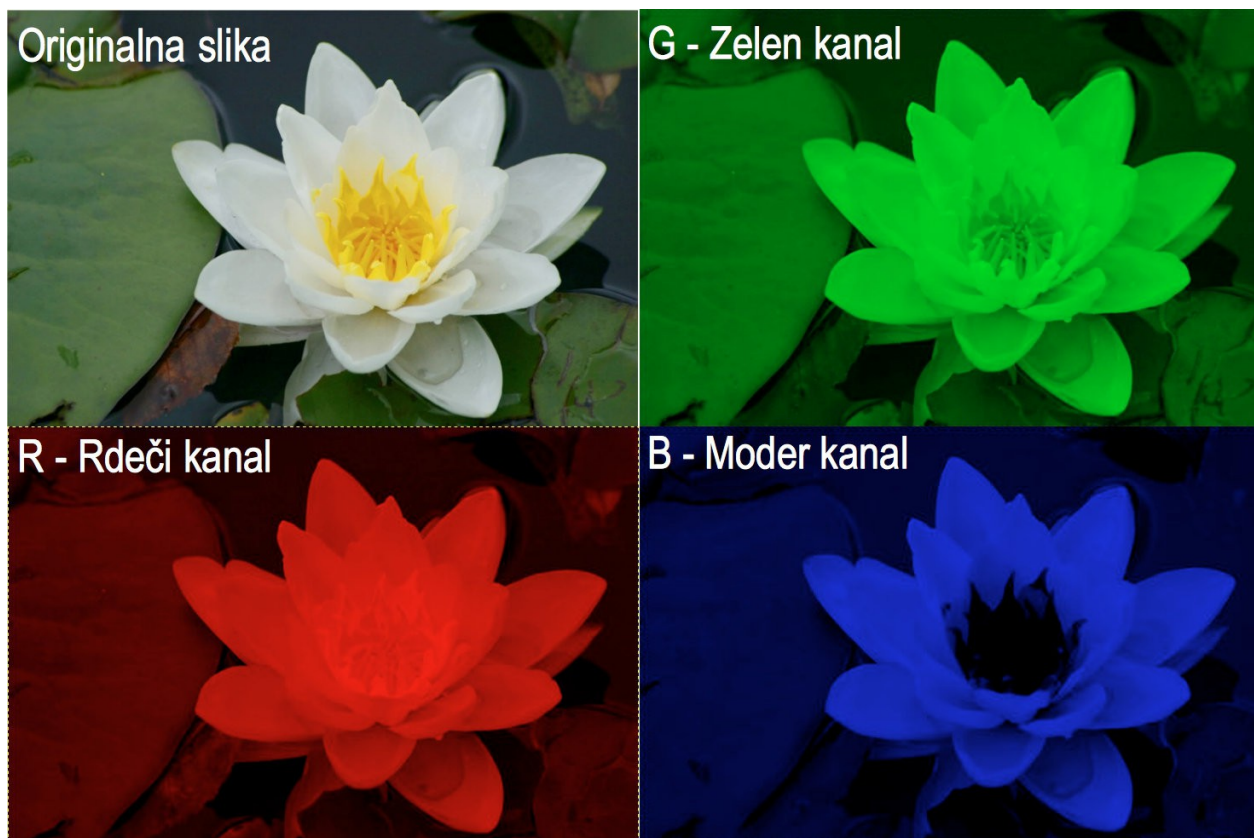
Uveljavitev tega modela so prinesle naslednje prednosti:

- manjše datoteke v primerjavi z drugimi modeli,
- dober prikaz na zaslonu (slika 123) glede na ostale barvne modele in večji barvni obseg kot pri modelu CMYK.



Slika 122: RGB-kocka, opazovana iz različnih pogledov (modelirano in upodobljeno v Blenderju)

Digitalna slika vsebuje tri kanale RGB (rdeči, zelen in moder - slika 123).



Slika 123: Predstavitev barvnih kanalov RGB v njihovi barvi (le zaradi ponazoritve)

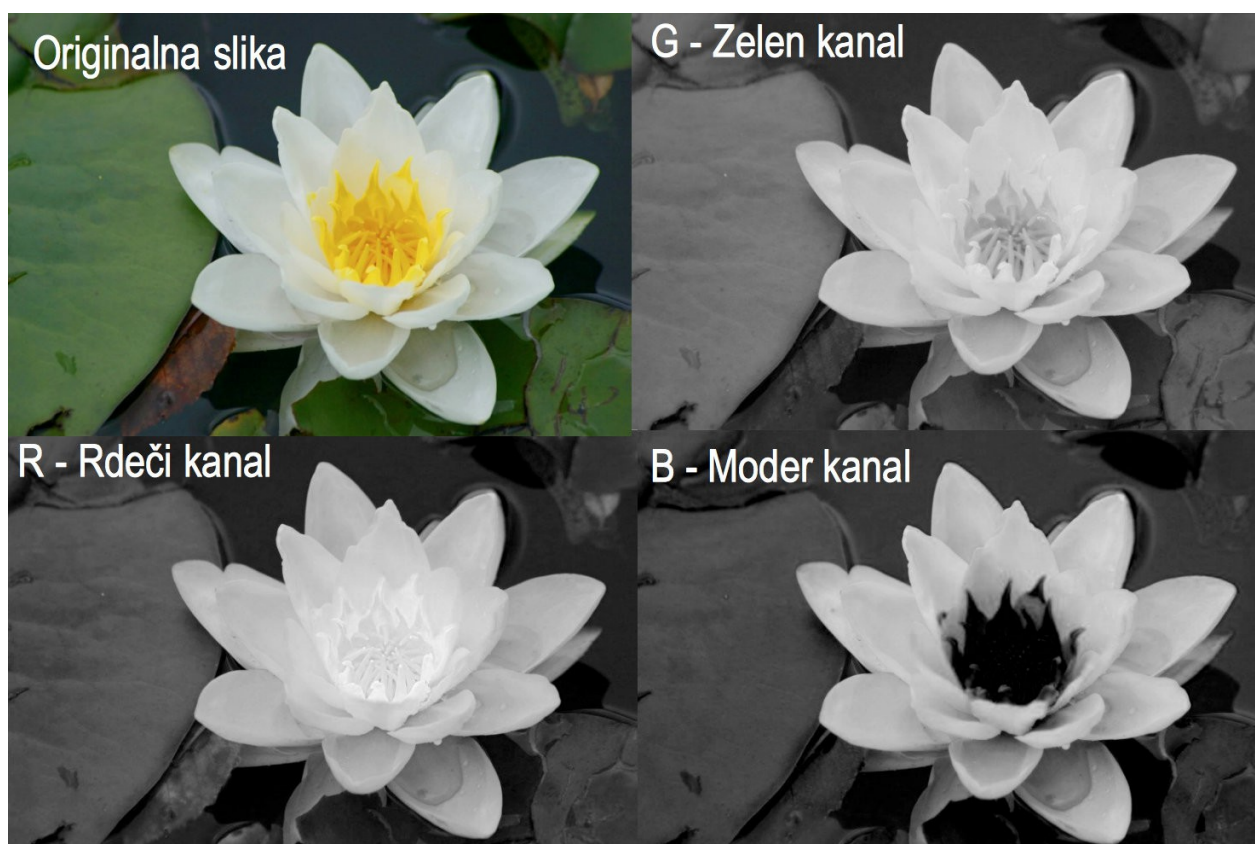
Na monitorju se sočasno prikazujejo vsi trije kanali RGB na svojih podtočkah (poglej naslov Prikaz slike na zaslonu, ki je na strani 91). Da si to lažje predstavljamo, je mogoče porabiti sani enega, dveh ali vseh treh kanalov pri maksimalni ali vrednosti nič (tabela 9).

Tabela 9: Nekaterne barve, ki nastanejo pri mešanju posameznega kanala ali vseh treh (pri vrednosti nič ali pri največ vrednosti FF, ki je desetiško 255)

Ime barve	R-Rdeča	G-Zelena	B-Modra
Črna	00	00	00
Rdeča	FF	00	00
Zelena	00	FF	00
Modra	00	00	FF
Rumena	FF	FF	00
Cian	00	FF	FF
Magentna	FF	00	FF

Bela	FF	FF	FF
------	----	----	----

Vrednosti kanalov RGB, kot jih vidimo na sliki 123, niso barvne slike, temveč številke od 0 do 255 (če bi opazovali vsak kanal posej, bi to bile sivinske informacije oz. le številke v datoteki) pri običajnem zapisu 8 bitov/kanal (slika 124). Če so vsi trije deleži kanalov enaki, je na tem delu slike videti bela barva, ko so vse tri vrednosti enake nič, je slika črna (vmesne vrednosti vseh treh kanalov pa so sivine). Ostale barve nastanejo z drugimi kombinacijami vrednosti slikovnih točk posameznih kanalov (pri 8-bitnem kanalu je to od 0 do 255, za 16-bitni je to od 0 do 65535). Na zaslonu kanali krmilijo vsako slikovno podtočko RGB in na ta način nastane barvna slika.



Slika 124: Originalna slika lotosa, v obliki slike so vidni ločeni sivinski RGB-kanali

Primeri nasičene rdeče barve lahko zapišemo na različne načine v modelu RGB (tabela 10).

Tabela 10: Nasičena rdeče barva s štirimi različnimi zapisi RGB

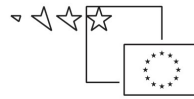
Zapis	RGB-trojka (angl. triplet)
Aritmetični	(1.0, 0.0, 0.0) (desetiško) – tega uporablja Blender
Odstotki	(100 %, 0 %, 0 %)
Digitalno 8-bitov na kanal	(255, 0, 0) (desetiško) ali šestnajstiško #FF0000
Digitalno 16-bitov na kanal	(65535, 0, 0) (desetiško)



KONZORCIJ ŠOLSKIH CENTROV



REPUBLIKA SLOVENIJA
MINISTRSTVO ZA ŠOLSTVO IN ŠPORT



Naložba v vašo prihodnost
OPERACIJO DELNO FINANCIRA EVROPSKA UNIJA
Evropski socialni sklad

Več o RGB-modelu lahko najdete na spletnih naslovih:

http://en.wikipedia.org/wiki/RGB_color_model, 16. 7. 2012

<http://de.wikipedia.org/wiki/RGB-Farbraum>, 16. 7. 2012

http://en.wikipedia.org/wiki/3D_displays, 16. 7. 2012

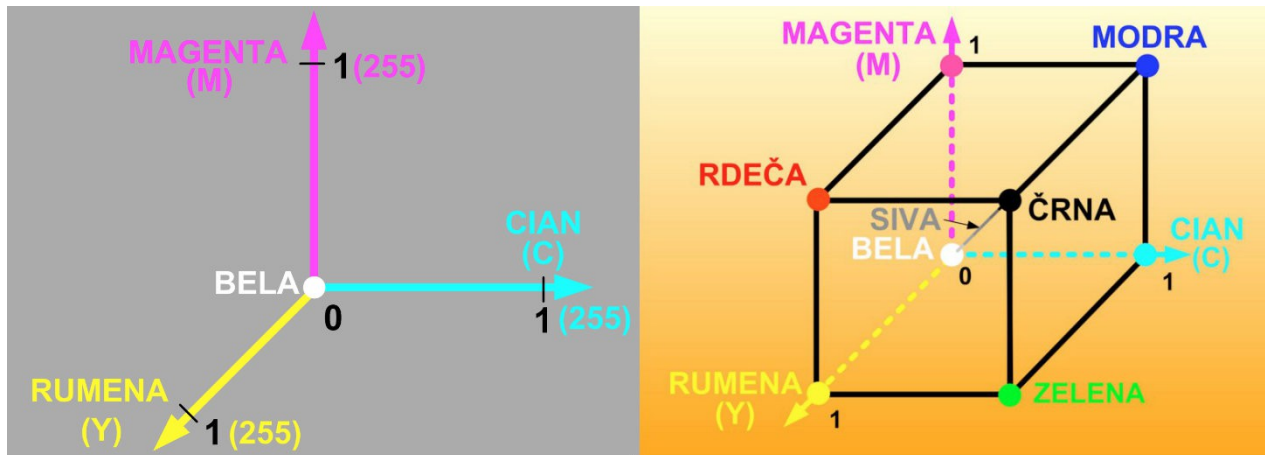
<http://en.wikipedia.org/wiki/Holography>, 16. 7. 2012

<http://en.wikipedia.org/wiki/Quattron>, 16. 7. 2012

<http://en.wikipedia.org/wiki/RGBA>, 16. 7. 2012

Barvni model CMYK

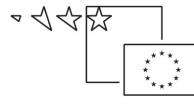
Je (odštevalni ali substraktivni) barvni model treh barv: cian (angl. Cyan), škrlatna ali magenta (angl. Magenta) in rumena (angl. Yellow) (slika 125 in 126). Če mešamo vse tri osnovne barve, dobimo umazano rjavo barvo. Zato se v kartuši tiskalnika uporabi namesto tega črna barva (angl. Black) in se uporablja t. i. barvno nadomeščanje (kar bo na sliki postalo čisto črno).



Slika 125: Trije pravokotni CMY-vektorji (levo) in idealna CMY-kocka (desno)

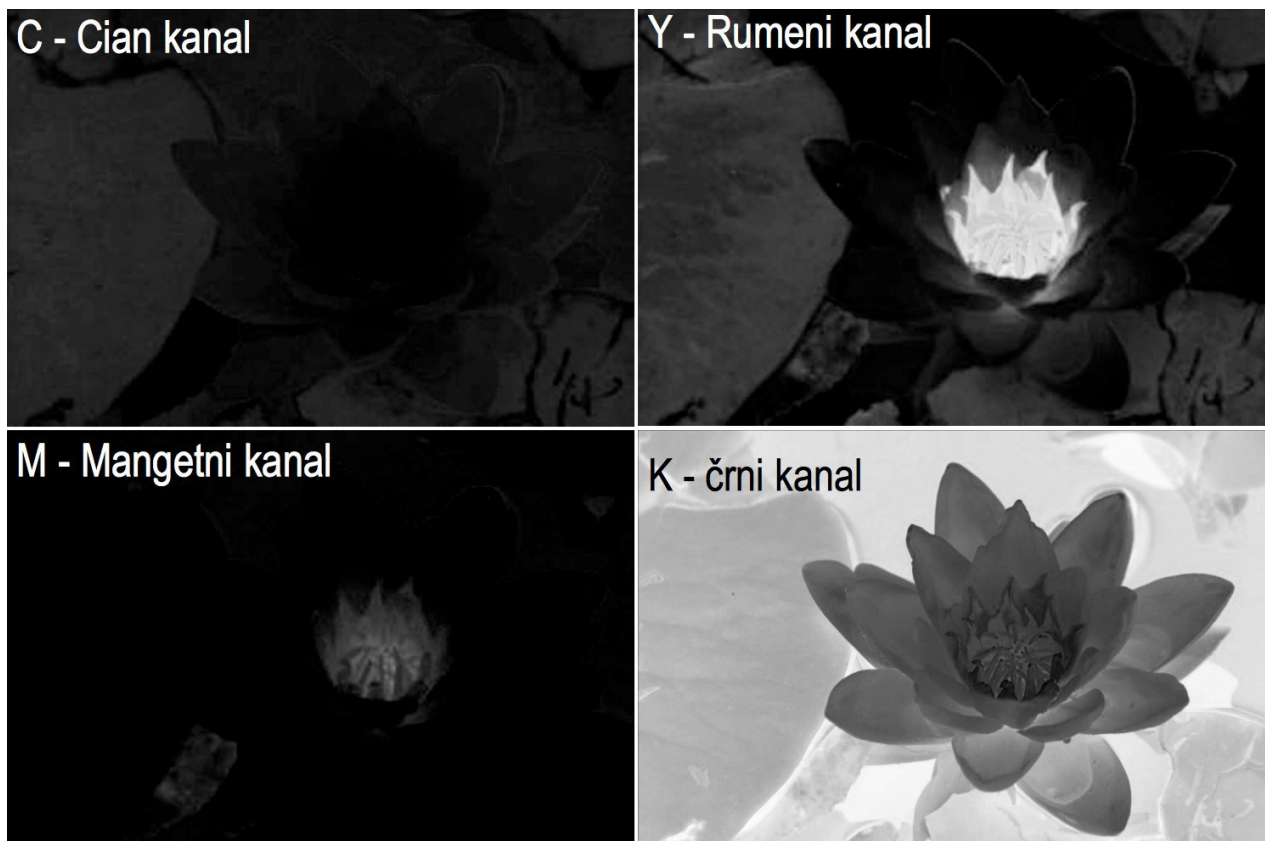
Ker je potrebno za črno vrednost natisa dobiti vrednosti, se ta model z RGB ne ujema popolnoma. Barvni obseg CMYK-a je dosti manjši kot RGB (ni sijočih barv) in to je odvisno od tiskalnika ali tiskarskega stroja (njegovega barvnega obsega).

CMYK je osnovni barvni model v tiskarstvu. Bela barva se posebej ne dobi z mešanjem treh komponent CMY, temveč je bela osnovna barva papirja.



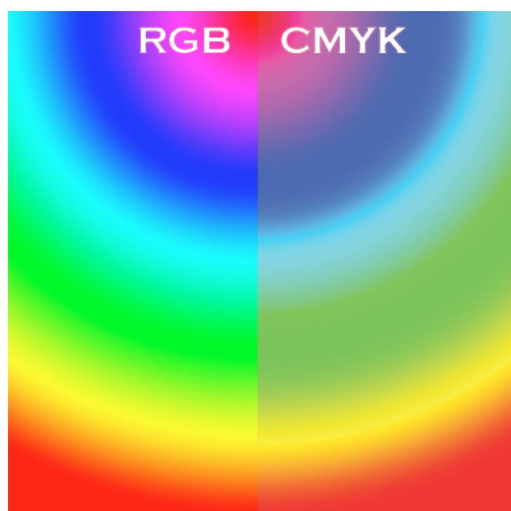
Slika 126: Predstavitev barvnih kanalov CMY v njihovi barvi (le zaradi ponazoritve)

Če bi bili pigmenti idealni (ali najbolj čisti), je dovolj, da barvni model CMYK vsebuje le štiri kanale (slika 127).



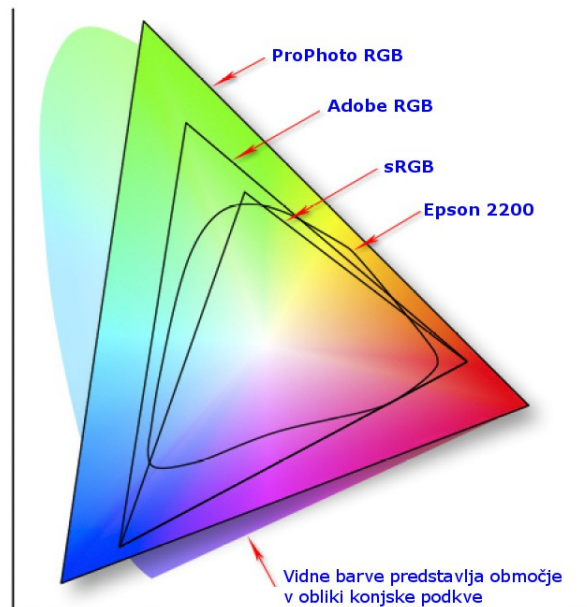
Slika 127: V obliki slike so razstavljeni sivinski CMYK-kanali iz začetne slike lotosa

Barvni model RGB je večji oz. vsebuje širši barvni obseg, kot je tiskalniški CMYK. Lepo se vidi, da ni tako bleščečih barv (slika 128). Pri seštevanju svetlob pri modelu RGB je zaradi sevanja svetlobe normalno, da so barve bolj bleščeče kot pri tisku. Tu se uporabijo le barvni pigmenti in sliko vidimo le zaradi odboja svetlobe zunanjega vira, kjer se žarek odbije od barve predmeta in potem potuje v oko opazovalca.



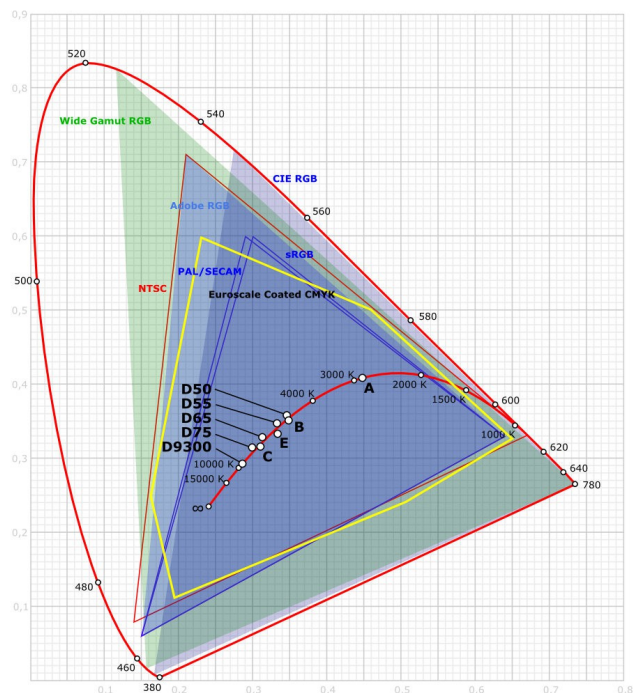
Slika 128: Primerjava v podobi barve med RGB- in CMYK-barvnim modelom

Barvni model sRGB (zmanjšan RGB), ki je standarden za operacijski sistem MS Windows, ima manjši barvni obseg, kot je CIE in NTSC. Na posameznih vogalih barvnega modela v CIE-diagramu se posamezni modeli ne ujemajo (slika 129).



Slika 129: Različna barvna obsega tiskalnika Epson 2200 normirane CIE-karakteristike standardnega opazovalca v obliki »podkve«, ki je v ozadju slike

Zanimiv je podoben diagram »podkve«, ki ima na sredini rdečo črto z označenimi kelvini sevanja črnega telesa. Na robu »podkve« so vrednosti valovnih dolžin v nm oz. na prejšnji sliki se tam vidijo barve (slika 130).



Slika 130: Standardna CIE RGB-karakteristika s sredinsko rdečo črto, kjer so označene temperature sevanja črnega telesa

Več o CMYK-modelu in drugih barvnih prostorih lahko najdete na spletnih naslovih:

http://en.wikipedia.org/wiki/CMYK_color_model, 16. 7. 2012

<http://en.wikipedia.org/wiki/CIELUV>, 16. 7. 2012

http://en.wikipedia.org/wiki/Color_space, 16. 7. 2012

<http://de.wikipedia.org/wiki/Farbraum>, 16. 7. 2012

<http://de.wikipedia.org/wiki/CIELUV-Farbraumssystem>, 16. 7. 2012

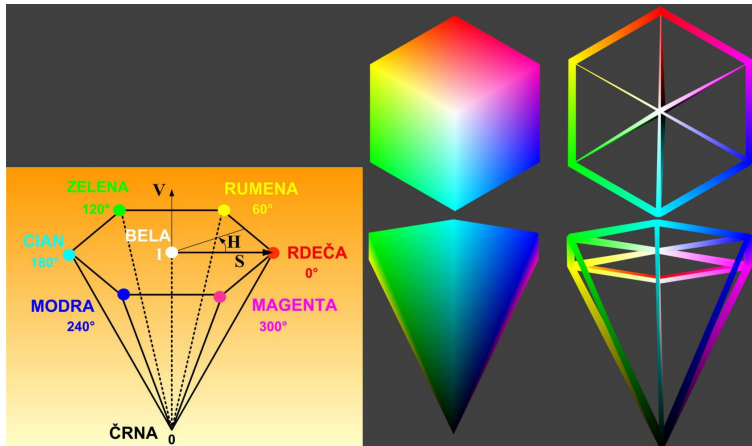
<http://de.wikipedia.org/wiki/LAB-Farbraum>, 16. 7. 2012

<http://de.wikipedia.org/wiki/CIE-Normvalenzsystem>, 16. 7. 2012

http://en.wikipedia.org/wiki/CIE_1931_color_space, 16. 7. 2012

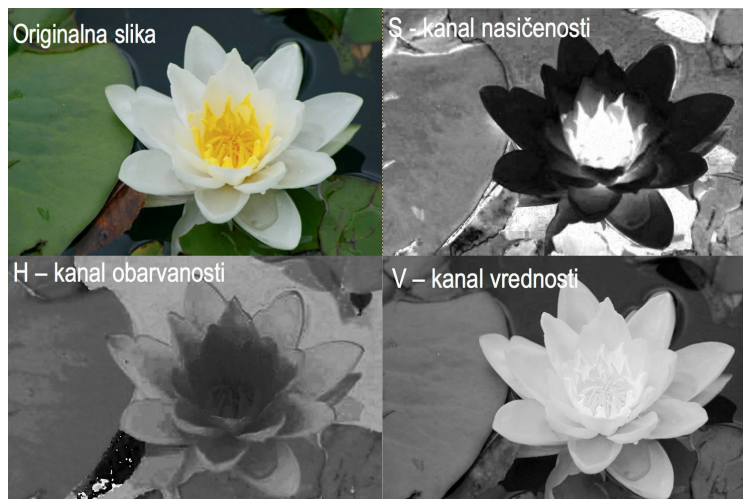
Barvni model HSV

Ta model, ki je bližje človeku, vsebuje tri kanale: barvni odtenek (angl. Hue), nasičenje (angl. Saturation) in svetlost (angl. Value). Osnova piramide je šestkotnik (slika 131), ki ima na diagonalnih vogalih komplementarne barvne pare (npr. rdeča-cian). Osnovna ploskev piramide ustreza vrednosti $V = 1$ in vsebuje relativno svetle barve. Barvni odtenek H se spreminja s kotom okoli navpične osi (0° pomeni rdečo). Vrednost nasičenosti S določa razdalja od osi V do plašča piramide (pri $S = 0$ imamo sive odtenke).



Slika 131: Šestkotnik HSV-modela

HSV-barvni model se najbolj približa vsakdanjem poimenovanju barv, kar se v modelu poimenuje odtenek (Hue), nasičenje (Saturation) - kako močna je barva (najbolj nasičena je na ploskvi) in svetlost (Value) - predstavlja višino do vrha (tam je črna akromatska vrednost - slika 131 in 132).



Slika 132: V obliki slike so razstavljeni sivinski HSV-kanali iz začetne slike lotosa

Heksadecimalni zapis barvnega prostor RGB

Skupina štirih bitov ali pol bajta (8 bitov) se imenuje **nibble** ali nybble. Ta enota se najbolj pogosto uporablja pri pretvorbi dvojiških v šestnajstiška števila, saj vzamemo 4 bitov in po pretvorbi dobimo eno šestnajstiško število. Na primer. $1101_{(2)} = 2^3 \cdot 1 + 2^2 \cdot 1 + 2^1 \cdot 0 + 2^0 \cdot 1 = 8 \cdot 1 + 4 \cdot 1 + 2 \cdot 0 + 1 \cdot 1 = 13_{(10)} = D_{(16)}$. Po vrsti je 13. cifra v šestnajstiškem sestavu črka D (ta predstavlja številko 13).

Za lažje pretvarjanje si lahko pomagamo s tabelo 11 (tudi za odčitavanje šestnajstiških vrednosti). Drugi primer pretvarjanja je za en bajt npr. $10101011_{(2)}$. Razdelimo ga na dva dela po štiri bite: $1010_{(2)}$ in $1011_{(2)}$ ter vsakega posebej pretvorimo in rezultat združimo po istem vrstnem redu, kot smo razstavili bajt. Torej: $1010_{(2)} = 2^3 \cdot 1 + 2^2 \cdot 0 + 2^1 \cdot 1 + 2^0 \cdot 0 = 8 \cdot 1 + 4 \cdot 0 + 2 \cdot 1 + 1 \cdot 0 = 10_{(10)} = A_{(16)}$ in $1011_{(2)} = 2^3 \cdot 1 + 2^2 \cdot 0 + 2^1 \cdot 1 + 2^0 \cdot 1 = 8 \cdot 1 + 4 \cdot 0 + 2 \cdot 1 + 1 \cdot 1 = 11_{(10)} = B_{(16)}$. Rezultat je $10101011_{(2)} = AB_{(16)}$.

Sami preverite naslednja dva primera:

104

Učno gradivo je nastalo v okviru projekta Munus 2. Njegovo izdajo je omogočilo sofinanciranje Evropskega socialnega sklada Evropske unije in Ministrstva za šolstvo in šport.

3. primer $10101001_{(2)} = X_{(16)} = \mathbf{A9}_{(16)}$

4. primer $11101101_{(2)} = X_{(16)} = \mathbf{ED}_{(16)}$

Tabela 11: Dvojiška in šestnajstiška števila za desetiška števila od 0 do 15 ali obratno

desetiško	0	1	2	3	4	5	6	7
dvojiško	0000	0001	0010	0011	0100	0101	0110	0111
šestnajstiško	0	1	2	3	4	5	6	7
desetiško	8	9	10	11	12	13	14	15
dvojiško	1000	1001	1010	1011	1100	1101	1110	1111
šestnajstiško	8	9	A	B	C	D	E	F

Iz standarda HTML 3.0 je bilo izbranih 16 barv iz palete Windows [VGA](#), ki je del sRGB (tabela 12).

Tabela 12: Šestnajst standardnih vrednosti barv

Barva	Ime	Trojček RGB	Rdeča	Zelena	Modra	Odtonek <u>Hue</u>	Nasičenost <u>Satur</u>	Svetlost <u>Light</u>
	White	#FFFFFF	100%	100%	100%	0°	0%	100%
	Silver	#C0C0C0	75%	75%	75%	0°	0%	75%
	Gray	#808080	50%	50%	50%	0°	0%	50%
	Black	#000000	0%	0%	0%	0°	0%	0%
	Red	#FF0000	100%	0%	0%	0°	100%	50%
	Maroon	#800000	50%	0%	0%	0°	100%	25%
	Yellow	#FFFF00	100%	100%	0%	60°	100%	50%
	Olive	#808000	50%	50%	0%	60°	100%	25%
	Lime	#00FF00	0%	100%	0%	120°	100%	50%
	Green	#008000	0%	50%	0%	120°	100%	25%
	Aqua	#00FFFF	0%	100%	100%	180°	100%	50%
	Teal	#008080	0%	50%	50%	180°	100%	25%
	Blue	#0000FF	0%	0%	100%	240°	100%	50%
	Navy	#000080	0%	0%	50%	240°	100%	25%
	Fuchsia	#FF00FF	100%	0%	100%	300°	100%	50%
	Purple	#800080	50%	0%	50%	300°	100%	25%

Po HTML4/CSS 1.0 so podana standardna imena in vrednosti za barve (tabela 13). Preostale barve in imena so objavljena na številnih spletnih straneh, ena od teh je tudi na wikipediji: http://en.wikipedia.org/wiki/Html_colors#HTML_color_names.

Tabela 13: Imena barv in kode za rdečo barvo po standardu HTML4/CSS 1.0

Ime <u>HTML</u>	Šestnajstiške kode			Desetiške kode		
	R	G	B	R	G	B
IndianRed	CD	5C	5C	205	92	92
LightCoral	F0	80	80	240	128	128
Salmon	FA	80	72	250	128	114
DarkSalmon	E9	96	7A	233	150	122
LightSalmon	FF	A0	7A	255	160	122
Red	FF	00	00	255	0	0
Crimson	DC	14	3C	220	20	60
FireBrick	B2	22	22	178	34	34
DarkRed	8B	00	00	139	0	0

Možno je tudi obratno pretvarjanje iz šestnajstiške vrednosti v desetiško. Poglejmo primer za RGB trojček šestnajstiške kode E9967A₍₁₆₎ in jo pretvorimo v desetiško število. To kodo razdelimo na tri dele tako, da vzamemo po dve cifri (dva nibbla), ki predstavljata en bajt oz. prvi je rdeč (R), drugi je zelen (G) in tretji je moder (B) kanal. Torej, E9 967A₍₁₆₎ = (R) E9 (G) 96 (B) 7A in vsakega zase pretvorimo v desetiško vrednost. Zapišimo najprej

$$(R) E9 (G) 96 (B) 7A = (R)16^1 \cdot E + 16^0 \cdot 9 (G) 16^1 \cdot 9 + 16^0 \cdot 6 (B) 16^1 \cdot 7 + 16^0 \cdot A$$

V tem računanju namesto šestnajstiških znakov vstavimo dvojiške vrednosti glede na tabelo 11. Potem pomnožimo vrednosti in seštejemo dve vrednosti vsakega kanala zase ter to zapišemo. Torej, (R) E9 (G) 96 (B) 7A = (R)16 · 14 + 1 · 9 (G)16 · 9 + 1 · 6 (B)16 · 7 + 1 · 10 = (R) 224 + 9 (G) 144 + 6 (B) 112 + 10

$$E9 967A_{(16)} = (R) 233_{(10)} (G) 150_{(10)} (B) 122_{(10)} = 233 | 150 | 122_{(10)}$$


Naloge za poglobljanje:

1. Na sliki 120 pogledaj zadnji oz. 3. del slike in odčitajte vrednost šestnajstiških kod ter določite barvo ter jo pretvorite v desetiško vrednost.
2. Iz tabele 13 izberi dve različni šestnajstiški in desetiški kodi in ju reši »pešč«. Rezultat lahko sam preveriš.
3. Pojdi na wikipedijin spletni naslov:
http://en.wikipedia.org/wiki/Html_colors#HTML_color_names in si od tam kopiraj vse barve X11 (angl. color names) in varne barve z ustreznimi imeni v datoteko urejevalnika besedil npr. LibreOffice. Iz teh barv izberi 10 najljubših in izpiši pripadajoče kode v RGB (desetiškem) in šestnajstiškem zapisu.

Ločljivost rastrske slike

Ločljivost (resolucija) rastrske slike je določena z zmnožkom števila slikovnih točk (angl. pixel) v vodoravni in navpični smeri (slika 133).

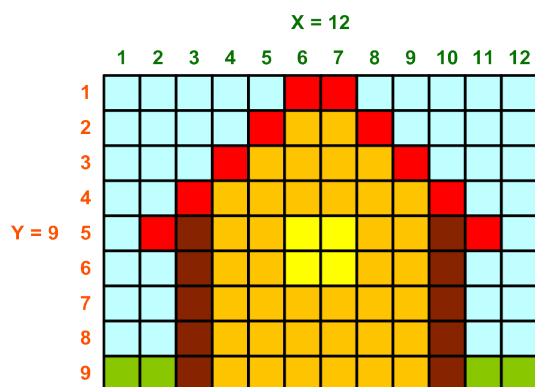
S formulo lahko izrazimo:

$$\text{Ločljivost slike} = \text{število slikovnih točk } x \cdot y. \quad (4.4.3)$$

Za zelo povečano sliko 133 hiše se ločljivost izračuna:

$$\text{Ločljivost slike} = 12 \cdot 9 \text{ (ali včasih kot } 12 \cdot 9 \text{)}.$$

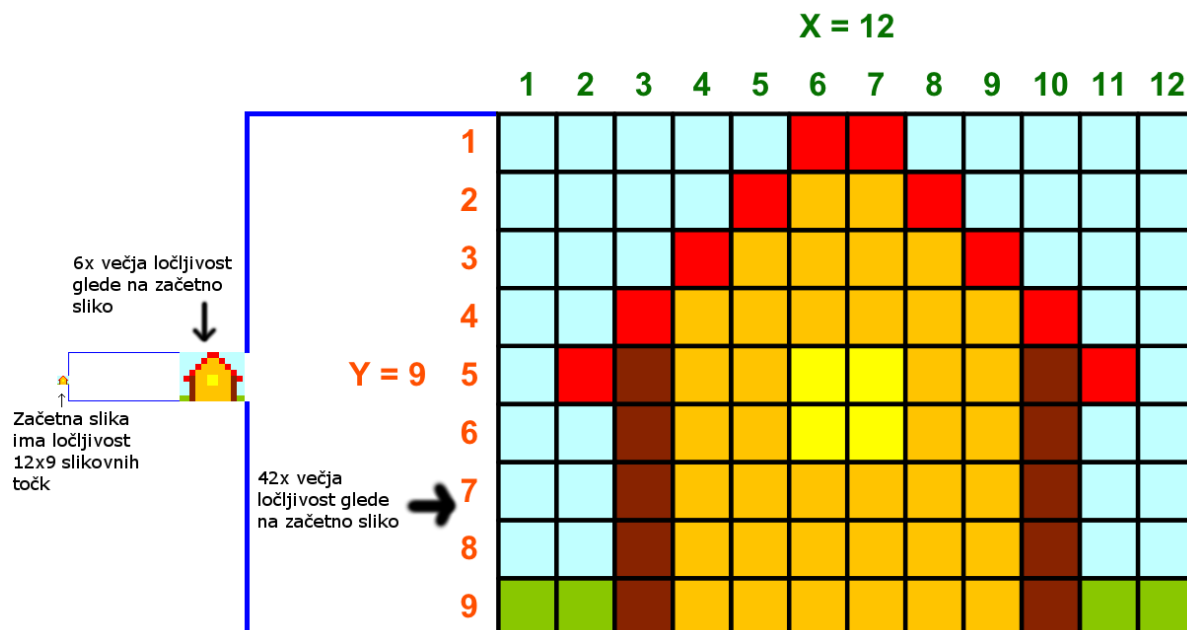
$$\text{Ločljivost slike} = 108 \text{ slikovnih točk.}$$



Slika 133: Slikovni primer za ponazoritev izračuna ločljivosti slike z velikostjo 12 x 9 slikovnih točk

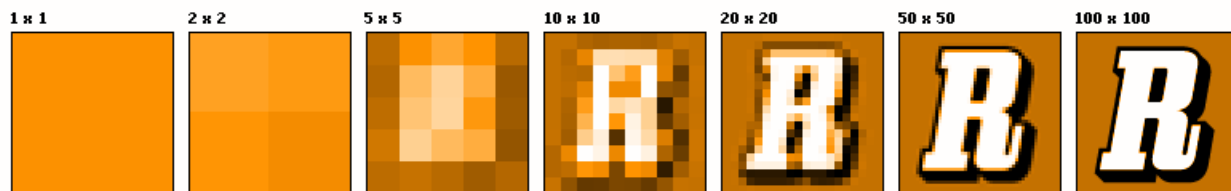
Začetna večja ločljivost posnete ali izračunane (upodobljene) slike omogoča njeno povečavo in izpis brez poslabšanja kakovosti oz. zmanjšanja ostrine. Ta slikovni primer slike hiše je vzet z namenom didaktične razlage.

Najbolje je, da se vzame ločljivost slike, ki je optimalna glede na uporabo in potem je ni potrebno skalirati (večati ali manjšati). Levo na sliki 134 se težko vidi, da je začetna slika z ločljivostjo 12 • 9 slikovnih točk predvsem mala in se njena vsebina bolj težko opazi (deluje bolj kot oranžna puščica kot slika hiše).



Slika 134: Od začetne slike 12 • 9 slikovnih točk do njene 42-kratne povečave

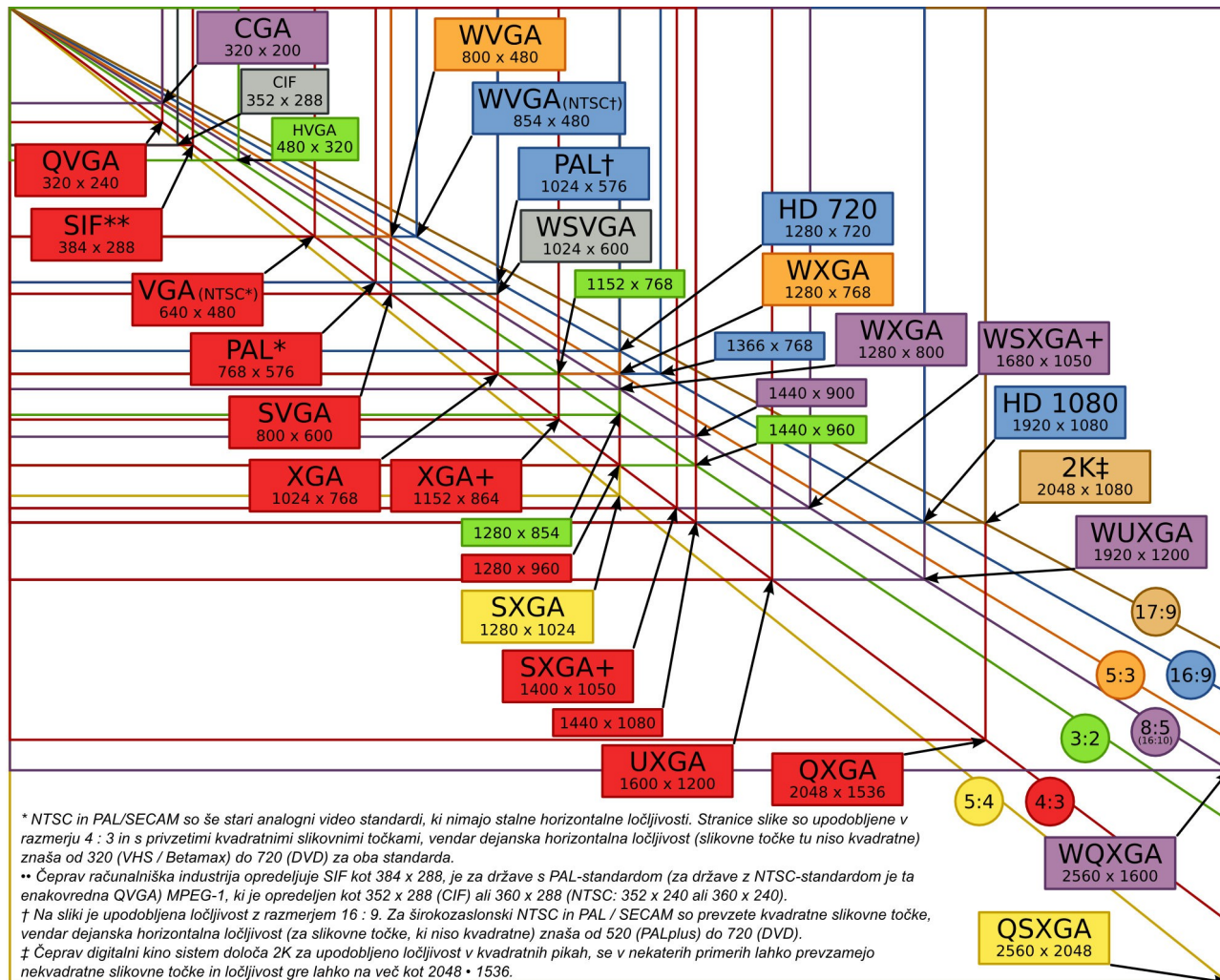
Za rastrske prikazovalnike in npr. za prikaz besedila je pomembna nastavljena ločljivost in manjša. Ob tej se bolj vidijo posamezne slikovne točke (slika 135).



Slika 135: Različne ločljivosti za črko R pri istih dimenzijah izhodiščnega kvadrata

Ločljivost zaslona ponavadi merimo v številu slikovnih točk na palec v enoti **PPI** (angl. pixels per inch).

Računalniški zasloni so glede ločljivosti segali od prvotnih 320 • 240 do današnjih najnaprednejših monitorjev z 2560 • 2048 slikovnih točk in razmerjem stranic 5 : 4 (slika 136).



Slika 136: Grafična ponazoritev razmerja stranic zaslonov in ločljivosti

Ločljivost današnjih računalniških zaslonov je zelo pestra in določena statistika iz ZDA je zbrana v naslednji tabeli 14.

Tabela 14: Običajne ločljivosti zaslonov v prvi polovici leta 2012 (vir <http://en.wikipedia.org/wiki/HDTV>)

Kratica	Razmerje stranic	Širina (px)	Višina (px)	% uporabnikov Steam storitve	% uporabnikov na spletu
VGA	4:3	640	480	0.02	n/a
SVGA	4:3	800	600	0.17	1.03
WSVGA	17:10	1024	600	0.31	2.25
XGA	4:3	1024	768	5.53	18.69
XGA+	4:3	1152	864	0.87	1.55
WXGA	16:9	1280	720	1.51	1.54
WXGA	5:3	1280	768	n/a	1.54
WXGA	16:10	1280	800	4.25	12.97
SXGA– (UVGA)	4:3	1280	960	0.72	0.72
SXGA	5:4	1280	1024	10.66	7.49
HD	~16:9	1360	768	2.36	2.28
HD	~16:9	1366	768	17.19	19.14
SXGA+	4:3	1400	1050	0.18	n/a
WXGA+	16:10	1440	900	7.60	6.61
HD+	16:9	1600	900	6.82	3.82
UXGA	4:3	1600	1200	0.53	n/a
WSXGA+	16:10	1680	1050	10.26	3.66
FHD	16:9	1920	1080	25.04	5.09
WUXGA	16:10	1920	1200	3.65	1.11
QWXGA	16:9	2048	1152	0.13	n/a
WQHD	16:9	2560	1440	0.72	0.36
WQXGA	16:10	2560	1600	0.19	n/a
	3:4	768	1024	n/a	1.93
	16:9	1093	614	n/a	0.63
	~16:9	1311	737	n/a	0.35
Drugi formati				1.29	7.25

Opomba

Statistika uporabnikov storitev omrežja Steam: anketiranci so sami oddali anketo o uporabi strojne opreme od januarja do maja 2012. [8]

Anketirani so bili izbrani med tremi milijoni obiskovalcev spletnih strani in potem porazdeljeni za preprečevanje lokacijske pristranskosti. [9] Številke v tabeli ne odražajo splošnega stanja uporabnikov računalnikov. Oznaka n/a pomeni, da ni odgovora oz. angl. not applicable, not available ali no answer.

Ločljivost naprav in medijev se je spreminjala z razvojem tehnike:

- Analogni in zgodnji digitalni
 - 352 × 240: [Video CD](#)
 - 300 × 480: [Umatic](#), [Betamax](#), [VHS](#), [Video8](#)
 - 350 × 480: Super Betamax, Betacam
 - 420 × 480: [LaserDisc](#), [Super VHS](#), [Hi8](#)
 - 640 × 480: Analogna televizija v ZDA ([NTSC](#))
 - 670 × 480: Enhanced Definition Betamax
 - 768 × 576: Analogna televizija v Evropi ([PAL](#), [SECAM](#))
- Digitalni
 - 720 × 480: [D-VHS](#), [DVD](#), [miniDV](#), [Digital8](#), Digital Betacam
 - 720 × 480: Widescreen DVD (anamorphic) širokokotni DVD
 - 1280 × 720: D-VHS, [HD DVD](#), [Blu-ray](#), HDV (miniDV)
 - 1440 × 1080: HDV (miniDV) videokamere
 - 1920 × 1080: HDV (miniDV), AVCHD, HD DVD, Blu-ray, HDCAM SR -HD-televizija
 - 1998 x 1080: 2K Flat (1.85:1)
 - 2048 × 1080: 2K Digital Cinema -2K digitani kino
 - 4096 × 2160: [4K Digital Cinema](#) -4K digitani kino
 - 7680 × 4320: [UHDTV](#)
 - Sekvence iz novejših filmov so optično prebrane na 2.000, 4.000 ali celo 8.000 stolpcev, zato se ti filmi imenujejo 2K, 4K ali 8K, in za kakovostne vizualne učinke, ki se urejajo na računalniku.
 - [IMAX](#), vključno z Imax HD in OMNIMAX: približno ločljivost 10.000 • 7000 (7000 vrstic). Gre za 70 Mpix (70 milijonov slikovnih točk), kar je trenutno najvišja ločljivost enega tipala digitalne kino kamere (podatek je iz januarja 2012).
- Film
 - Klasični [35 mm filmski](#) trak je optično prebran za distribucijo na DVD-jih na 1080 ali 2000 linij, kot je bilo leta 2005.
 - Dejanska ločljivost kamere s prvotnim negativnim 35 mm filmom je predmet mnogih razprav. Izmerjene ločljivosti negativnih filmov so v razponu od 25 - 200 lp /mm, kar ustreza obsegu 325 linij za kamero Techniscope in (teoretično) več kot 2300 vrstic za filmski trak z imenom 4-perf, uporabljen za posnetke na T-Max 100. Arhivarji se splošno strinjajo, da je 35 mm film optično prebran na 4 K več kot primerna za arhivske namene. [6]
- Za tiskanje so v nadaljevanju podani primeri v več tabelah.

Ločljivosti za tiskanje so različne in velikosti slik so prav tako odvisne od začetnega števila slikovnih točk in same nastavitve ločljivosti tiskanja DPI (tabela 15).

Tabela 15: Sprememba velikost slike ob spremembi DPI in pri konstantnih 1000 slikovnih točkah

DPI	Število slikovnih točk (Pixels)	cm
800	1000	3,18
300	1000	8,47
200	1000	12,7
72	1000	35,28

Če želimo tiskati sliko kvadratne velikosti s stranico 10 cm in tiskati z različnimi nastavitvami DPI, je potrebno imeti različno število slikovnih točk (tabela 16).

Tabela 16: Velikost slike je na stranico 10 cm, kar dosežemo ob spreminjanju ločljivosti in DPI

DPI	Število slikovnih točk (Pixels)	cm
800	3150	10
300	1181	10
200	787	10
72	283	10

Za klasično tiskanje s 300 DPI pri standardnih A-formatih papirja je potrebno imeti slike z ustreznim številom slikovnih točk (tabela 17).

Tabela 17: Potrebno število slikovnih točk za doseganje tiskanja s 300 DPI pri standardnih A-formatih papirja

DPI	Število slikovnih točk (Pixels)	Mpx (milijon slik. točk)	mm	Velikost papirja
300	11114x14008	155,68	941x1186	A0
300	7016x11114	77,976	594x941	A1
300	4961x7016	34,806	420x594	A2
300	3508x4961	17,403	297x420	A3
300	2480x3508	8,702	210x297	A4
300	1748x2480	4,351	148x210	A5
300	1240x1748	2,175	105x148	A6
300	874x1240	1,083	74x105	A7
300	614x874	0,541	52x74	A8

Več o CMYK-modelu in drugih barvnih prostorih lahko najdete na spletnih naslovih:

http://en.wikipedia.org/wiki/Paper_size, 16. 7. 2012

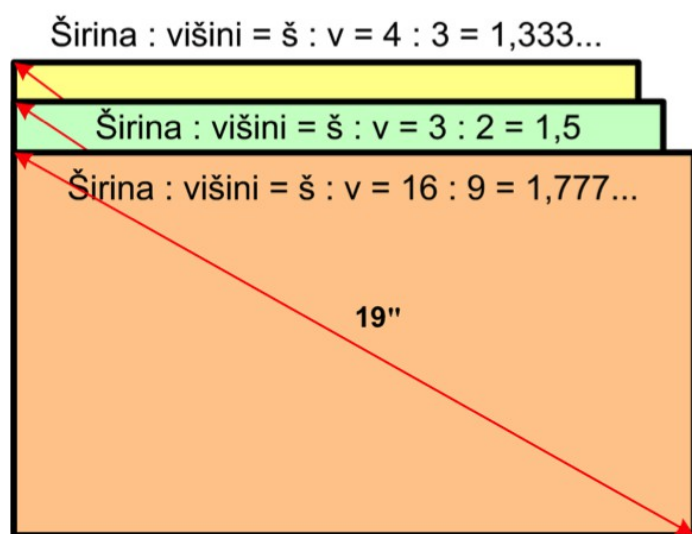
http://en.wikipedia.org/wiki/Negative_pull_down, 16. 7. 2012

http://en.wikipedia.org/wiki/2-perf_pull_down, 16. 7. 2012

Diagonale zaslonov (monitorjev) in razmerja stranic

Klasični računalniški zasloni ali monitorji so imeli razmerje zaslona 4 : 3 med številom slikovnih točk v vodoravni in navpični osi. Danes imajo monitorji, zasloni in slika iz projektorja eno razmerje med stranicami: 17 : 10, 5 : 3, 16 : 10, 16 : 9, 4 : 3, 5 : 4. Konkretno si pogledjmo en primer za 19" (slika 137).

V vseh treh pravokotnikih je diagonala zaslona enaka 19 " in različno je razmerje stranic za širino in višino. Pri razmerju 4 : 3 in 3 : 2 so monitorji enake površine, le pri 16 : 9 je površina za 7,42 % manjša glede na prejšnji dve razmerji. Danes najpopularnejši monitorji z razmerjem 16 : 9 so širši, kar je ugodneje glede na vidno človeško polje.



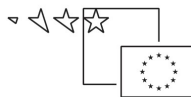
Slika 137: Primerjava med tremi razmerji stranic zaslonov z diagonalo 19"

V tabeli 18 so različne diagonale preračunane v širino in višino za posamezno razmerje stranic zaslona.

Tabela 18: Različne diagonale in razmerja stranic

Diagonala	Razmerje 4 : 3			Razmerje 3 : 2		Razmerje 16 : 9	
	D (cm)	Š (cm)	V (cm)	Š (cm)	V (cm)	Š (cm)	V (cm)
15	38,1	30,48	22,86	31,70	21,13	33,21	18,68
17	43,18	34,54	25,91	35,93	23,95	37,63	21,17
19	48,26	38,61	28,96	40,15	26,77	42,06	23,66
21	53,34	42,67	32,00	44,38	29,59	46,49	26,15
23	58,42	46,74	35,05	48,61	32,41	50,92	28,64
25	63,5	50,80	38,10	52,84	35,22	55,35	31,13

4Z D je označena diagonala monitorja v palcih ali inčih ter v centimetrih. V označuje višino monitorja, Š pa njegovo širino v cm.

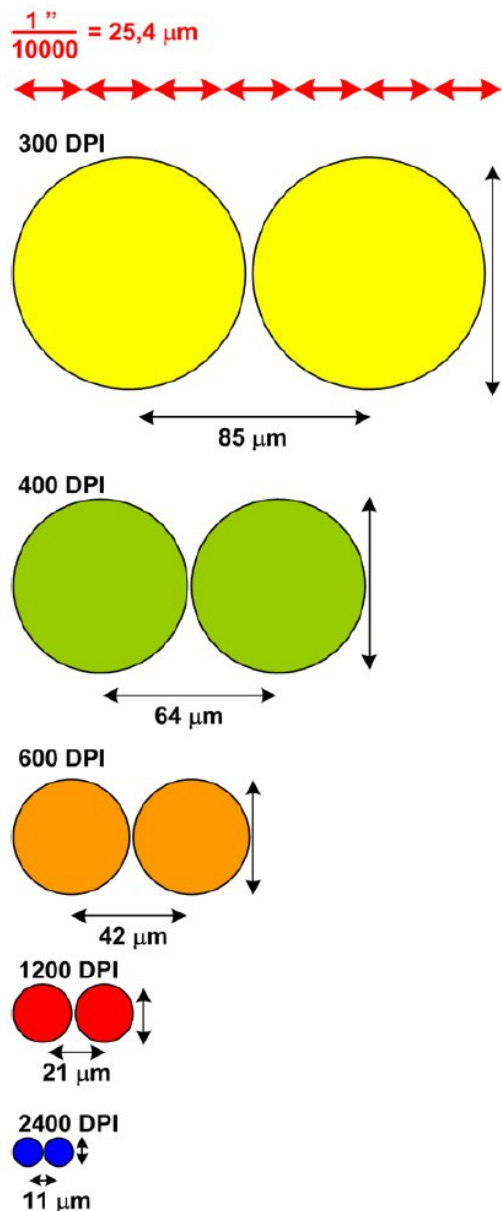


27	68,58	54,86	41,15	57,06	38,04	59,77	33,62
29	73,66	58,93	44,20	61,29	40,86	64,20	36,11
31	78,74	62,99	47,24	65,52	43,68	68,63	38,60
33	83,82	67,06	50,29	69,74	46,49	73,06	41,09

Spreminjanje ločljivosti zaslona

Ločljivost zaslona merimo v številu slikovnih točk na palec v enoti PPI (angl. pixels per inch) ali DPI.

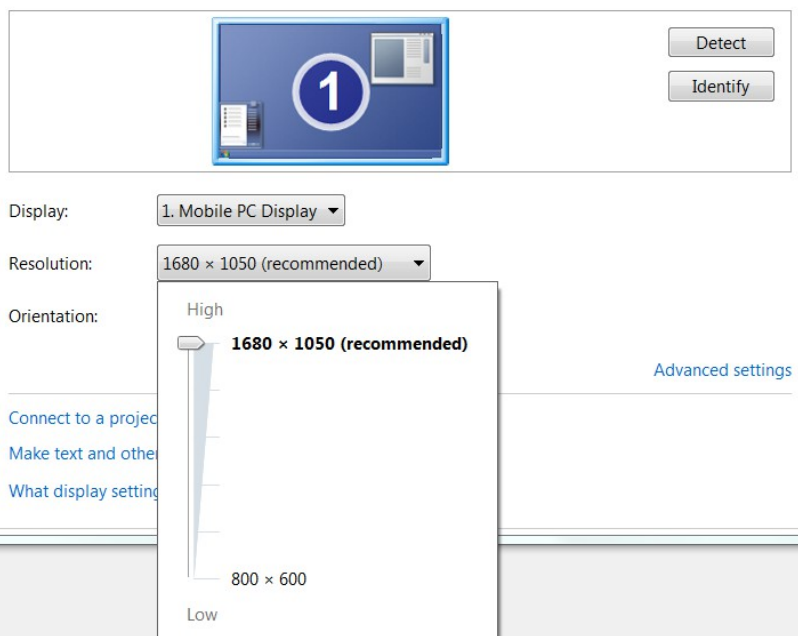
Računalniki MAC so imeli na začetku ločljivost zaslona 72 PPI, na osebnih računalnikih je ponavadi nastavljena 96 PPI, 120 PPI ali več (slika 138). Čim višje je nastavljena ločljivost zaslona, tem manjši sta prikazani grafika in pisava.



Slika 138: Nastavljena DPI-ločljivost vpliva na velikost slikovne pike.

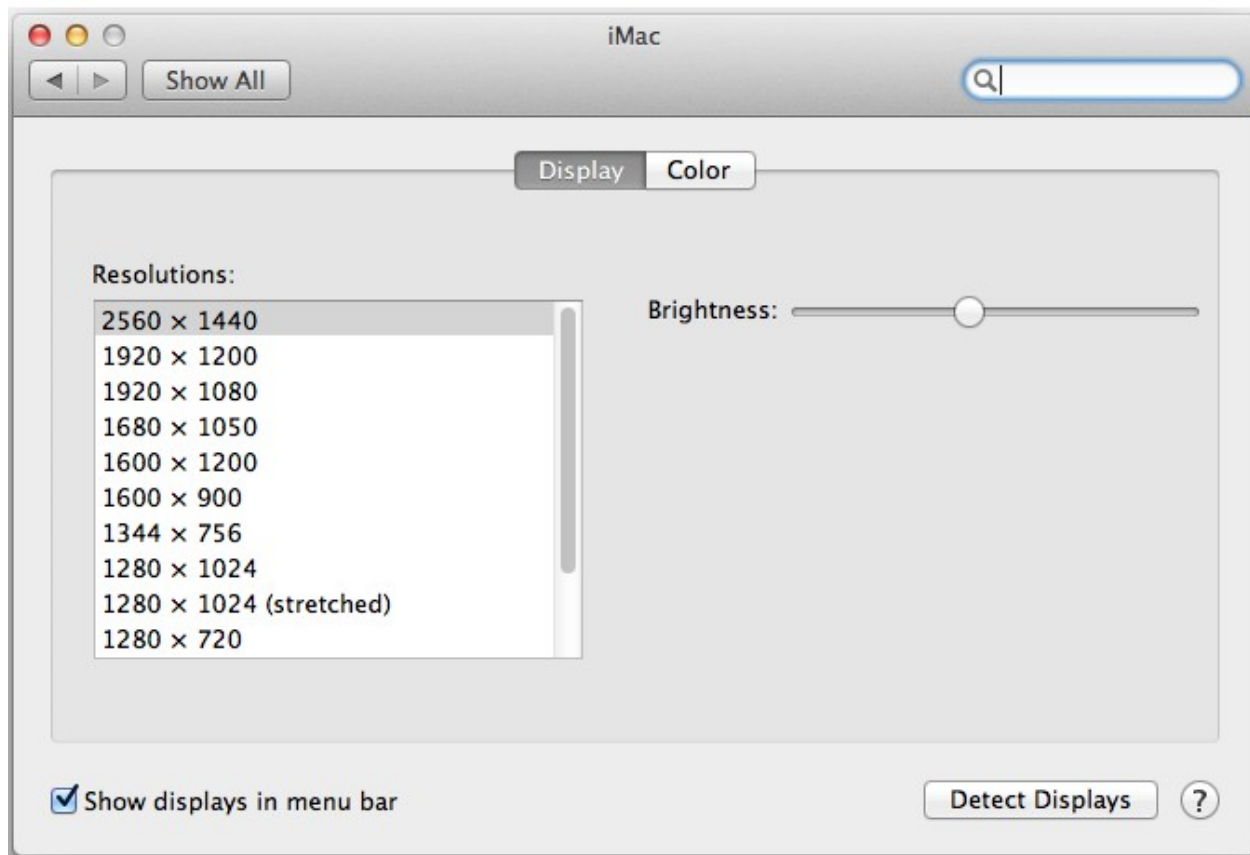
Na vsakem zaslonu je možno nastaviti različne ločljivosti, pri LCD-jih je ponavadi s strani proizvajalca zaslona ena ločljivost priporočljiva. Prav tista, ki fizično sestavlja zaslonsko matriko in zato je slika tu najbolj ostra (slika 139).

Change the appearance of your display



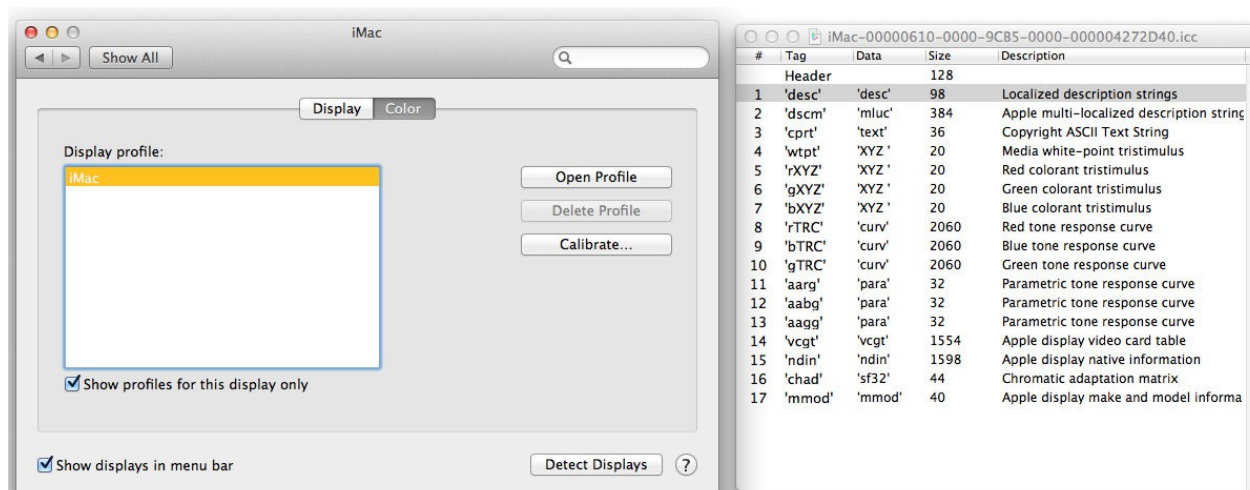
Slika 139: Sprememba ločljivosti na monitorju pri OS Windows 7

Vsaj operacijski sistem ima na podoben način nastavitve za spreminjanje ločljivosti, kot npr. MAC OS X (slika 140).



Slika 140: Sprememba ločljivosti zaslona Apple-ovega operacijskega sistema MAC OS X

V prejšnji sliki ob kliku na gumb Color preklpimo v barvno upravljanje zaslona ter v barvne profile icc (slika 141).

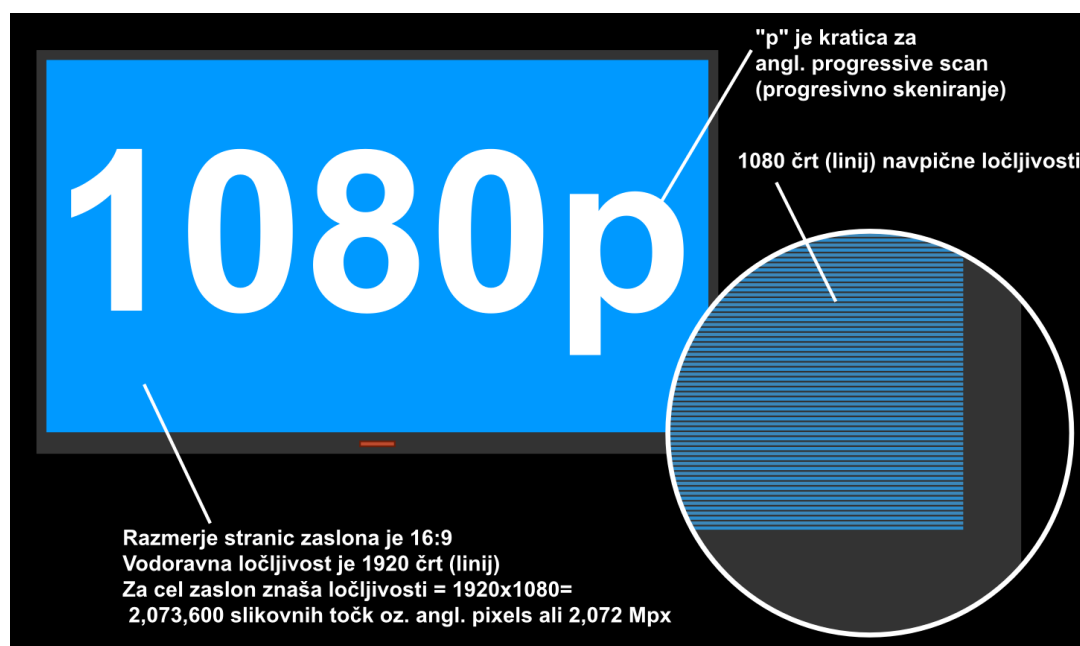


Slika 141: Na desni sliki pogledamo z gumbom Open Profile (Odpri profil) barvni profil, ki je na desni s posameznimi parametri.

Ločljivost za HDTV

Zaradi relativno »novega« standarda digitalne televizije visoke ločljivosti t. i. HDTV (angl. High-definition television) je postal popularen tudi zaslon z razmerjem stranic 16 : 9 (slika 142) ali 16 : 10.

Ločljivost takega zaslona je 1920 • 1080 slikovnih točk, kar znaša 2 072 736 slikovnih točk (okrog 2 milijona oz. ~ 2 Mp ali ~ 2 Mpx).



Slika 142: Slikovno pojasnilo ločljivosti HDTV

V tabeli 19 so navedene ločljivosti HDTV, število slikovnih točk, razmerje stranic zaslona in razmerje stranic slikovne točke.

Tabela 19: Različne ločljivosti sprejemnikov HDTV

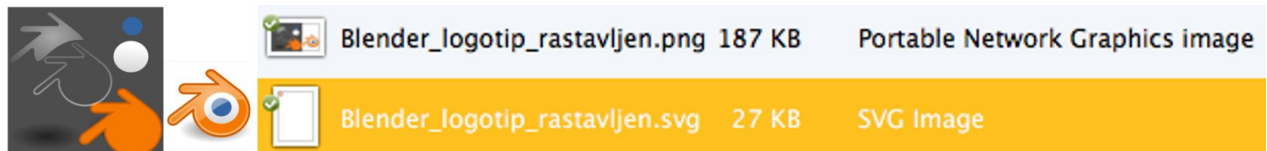
Podpora formata	video	Naravna ločljivost (š×v)	Št. slikovnih točk -piksli (Megapiksli)	Razmerje (X:Y)	Točke
				Slika	Točke
720p		1024×768	786,432 (0.8)	16:9	4:3
1280×720		XGA			
		1280×720	921,600 (0.9)	16:9	1:1
		1366×768 WXGA	1,049,088 (1.0)	683:384 (približno 16:9)	približno 1:1
1080i ⁵		1280×1080	1,382,400 (1.4)	32:27 (približno 16:9)	3:2
1920 × 1080					
1080p ⁶		1920×1080	2,073,600 (2.1)	16:9	1:1

5 Angl. Interlace ali prepletanje pomeni izris slike tako najprej lihe in potem sodih vrstic.

1920×1080				
2160p 3840×2160	3840×2160	8,294,400 (8.3)	16:9	1:1

Velikost digitalne slike

Velikost slikovne datoteke je odvisna vsaj od ločljivosti, barvne globine in izbranega formata za shranjevanje. V nadaljevanju sledi preprosta razlaga, kako približno nekateri računalniški programi za obdelavo rastrskih slik uporabljajo algoritme in zapisuje slikovne datoteke. Velikost datoteke rastrske slike (fotografije) je ponavadi dosti večja od vektorske (risbe so ponavadi precej preproste - slika 143). Pri animaciji se uporabljajo različne tehnike senčenja, da dobimo foto realističen prikaz. Zaradi tega se računalniško upodobljeni objekti danes težko ločijo od fotografskih posnetkov.



Slika 143: Primerjava velikosti iste slike v rastrskem PNG- in vektorskem SVG-formatu

Osnovno velikost digitalne slike dobimo z zmnožkom ločljivosti in barvne globine. Za določanje števila zlogov (bajtov ali bytov) je potrebno prejšnjo vrednost deliti z 8 (1 zlog ima 8 bitov). Velikosti datoteke dodajo še po en bajt za zapis števila točk za vodoravno ločljivost 12 točk in enega za zapis vertikalne ločljivosti 9 točk. Če bi zapisovali to sliko in jo potem prikazovali, bi isto ločljivost imela slika dimenzij: 1 x 108, 2 x 54, 3 x 36, 4 x 27 in 6 x 18 (seveda lahko mesto zamenjamo z obema številčkama in dobimo njuno pokončno postavitev). Slika bo imela tudi ime in zaradi tega dodajmo še 14 bajtov za zapis imena (osem znakov - 8 bajtov) in končnice npr. .png vključno z decimalno piko (npr. .bmp (s piko) torej 4 znaki oz. 4 bajti).

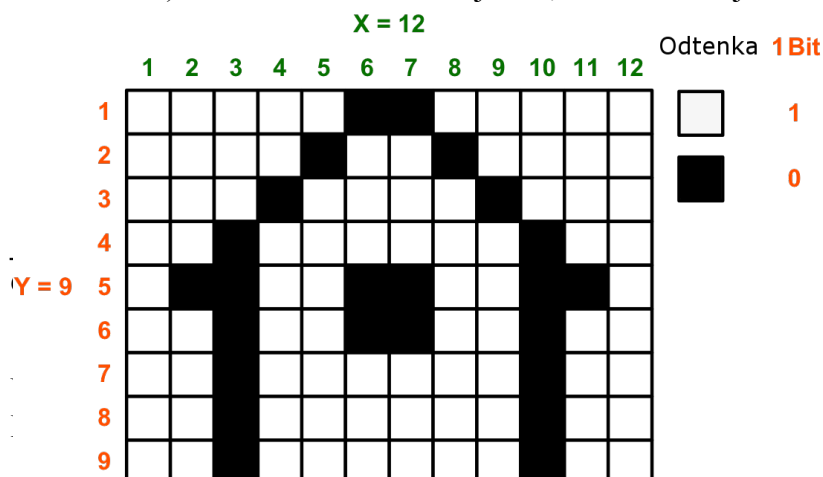
Primer: **Velikost ČB slike (datoteke) = 12 • 9 • barvna globina (ali 12 • 9 • barvna globina) + 1 B (za širino slike - 12 točk) + 1 B (za višino slike - 9 točk) + 12 bajtov (za ime + tri črkovne končnice, npr. moj_slik.bmp je 12 znakov).**

Ločljivost slike = 108 točk; pri barvni globini 1 bit je potrebno imeti 108 • 1b + 14 B • 8 b = 220 bitov za zapis slike.

Pretvorimo to še v število bajtov (angl. Byte), če vemo, da tega sestavlja osem bitov.

Št. zlogov = št. bitov / 8 = 220 / 8 = 27,5 zlogov, kar zaokrožimo na celih 28 zlogov = 28 B.

Če bi bila velikost slike v eno smer več kot 255 točk, bi morali za to smer vzeti 2 B (do velikosti slike 65 535). Če bi bili obe dimenziji taki, bi bila ločljivost te slike 4 294 967 296 slikovnih



Slika 144: Črno-bela slika hiše

ot angl. Interlace).

ilo sofinanciranje

točk (mislim, da takega tipala še ni na naši Zemlji)! V Blenderju je omejena velikost upodabljanja na 10 000 • 10 000 slikovnih točk, kar znaša 100 000 000 (sto milijonov slikovnih točk).

Velikost sivinske digitalne slike

Sivinska slika se običajno med ljudmi imenuje črno-bela (kar ni prav, saj vsebuje več svin kot le skrajni akromatski vrednosti). Na spodnji sliki je 6 sivinskih odtenkov (slika 145), za njihov zapis potrebujemo vsaj tri bite ($2^3 = 8$ odtenkov, dva sta odvečna oz. redundantna).

Vrednosti posameznih 6 sivinskih odtenkov je potrebno dodatno zapisati, vsakega vsaj z enim zlogom. V ta namen rabimo: $6 \cdot 8 B = 48 B = 48 \cdot 8 b = 384 b$. Za sivinske vrednosti je značilno, da imajo vsi trije RGB-kanali isto število. Če bi se razlikovali le za eno številko, to ne bi bila več sivina. Npr. trojček R G B = 66 67 66 je za eno število več na zelenem kanalu - sicer bi to težko opazili na prosto oko.

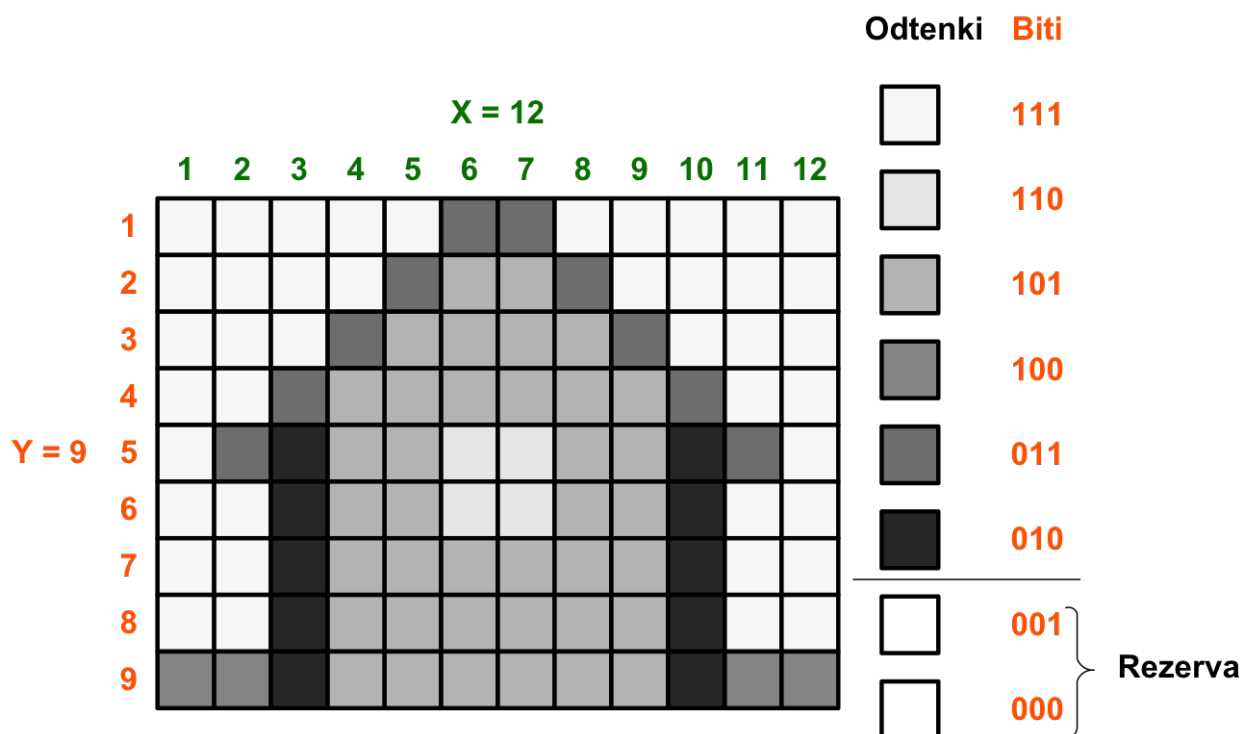
Primer izračuna velikosti datoteke s sivinsko sliko iz 6 odtenkov:

Velikost sivinske slike = št. slikovnih točk • št. bitov + biti za zapis svin + 1 B (za širina slike - 12 točk) + 1 B (za višino slike - 9 točk) + 12 bajtov (za ime + tri črkovne končnice, npr. moj_slik.bmp je 12 znakov).

Velikost slike = $108 \cdot 3 + 384 + 14 \cdot 8 = 324 b + 384 b + 112 b = 820 b$.

Pretvorimo število bitov v bajte (angl. Byte), če vemo, da tega sestavlja osem bitov.

Št. zlogov = velikost / 8 b = $820 / 8 = 102,5$ zlogov, kar zaokrožimo na 103 zlogov = **103 B**. Rezultat pove, da je črno-bela slika, ki je imela 28 B, za kar **3,66 večja od sivinske slike!**



Slika 145: Sivinska digitalna slika in pripadajoča sivinska paleta (sivinski odtenki so kodirani)

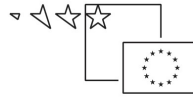
Učno gradivo je nastalo v okviru projekta Munus 2. Njegovo izdajo je omogočilo sofinanciranje Evropskega socialnega sklada Evropske unije in Ministrstva za šolstvo in šport.



KONZORCIJ ŠOLSKIH CENTROV



REPUBLIKA SLOVENIJA
MINISTRSTVO ZA ŠOLSTVO IN ŠPORT



Naložba v vašo prihodnost
OPERACIJO DELNO FINANCIRA EVROPSKA UNIJA
Evropski socialni sklad

Velikost barvne indeksirane digitalne slike

Na spodnji sliki je 6 barvnih odtenkov. Za zapis le-teh je potrebno imeti vsaj tri bite ($2^3 = 8$ odtenkov, dva sta odvečna oz. redundantna) (slika 146). Določanje, kateri odtenki so na sliki in prirejanje indeksov le-tem (kombinacije bitov), se imenuje indeksiranje. Algoritem procesiranja se prične npr. pri shranjevanju slike v slikovnem formatu GIF (angl. Graphics Interchange Format).

Tudi vrednosti posameznih 6 barv je potrebno dodatno zapisati, vsako vsaj s tremi zlogi $3 \cdot 8 = 24$.

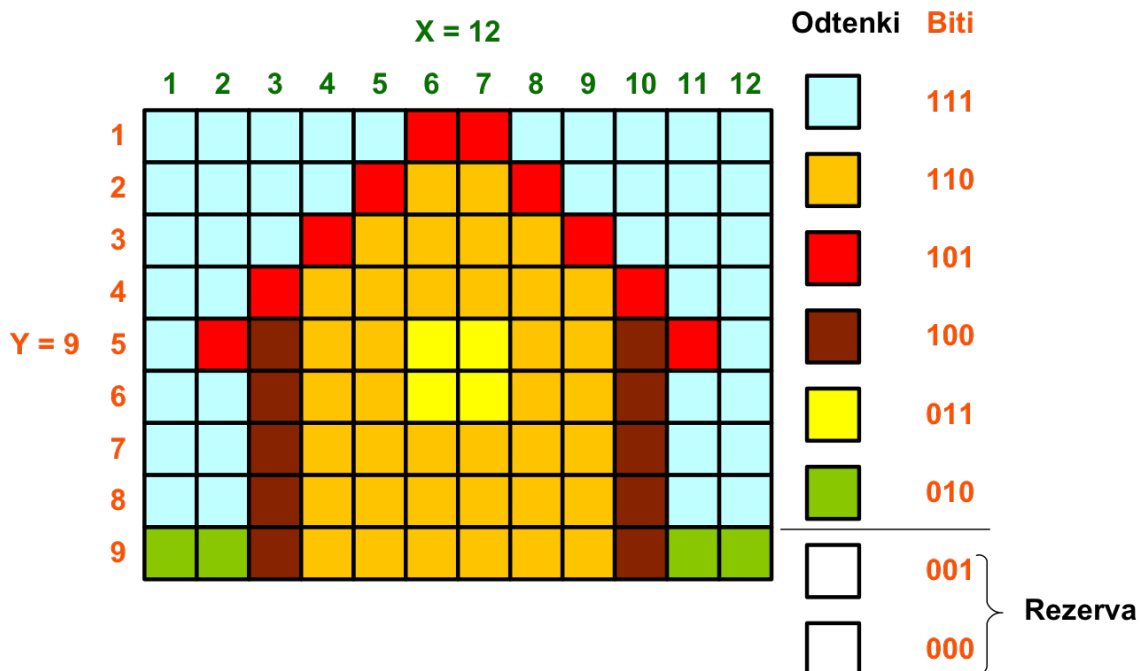
Skupaj je torej za zapis barv potrebno: $3 \cdot 8 \cdot 6 \text{ B} = 24 \cdot 6 \text{ B} = 144 \text{ B} = 144 \cdot 8 \text{ b} = 1152 \text{ b}$.

Osnovni zapis velikosti slike je št. točk \cdot št. bitov = $108 \cdot 3 \text{ b} = 324 \text{ b} / 8 = 40,5 \text{ B}$.

Primer: **Velikost indeksirane barvne slike = št. točk \cdot št. bitov/8 + kodiranje barv + 1 B (za širino slike - 12 točk) + 1 B (za višino slike - 9 točk) + 12 bajtov (za ime + tri črkovne končnice, npr. moj_slik.gif je 12 znakov) = $144 \text{ B} + 40,5 \text{ B} + 14 \text{ B} = 198,5 \text{ B}$.**

Slika bo velika 198,5 zlogov, kar zaokrožimo na 199 zlogov = 199 B.

Če primerjamo to velikost s sivinsko, ki ima 103 B, je sedaj indeksirana skoraj dvakrat večja. V primerjavi s črno-belo, ki ima 28 B, je indeksirana za 7,1 krat večja. Seveda, če bi bilo več barv npr. 102, bi potrebovali indeks s 7 biti, saj je $2^7 = 128$. Potem bi imeli 26 odvečnih ali redundantnih mest za nove barve ali prosojnost. Ista slika bi bila potem velikosti $3 \cdot 8 \cdot 102 \text{ B} + 94,5 \text{ B} (108 \cdot 7/8 \text{ B}) + 14 \text{ B} = 2556,5 \text{ B} = 2557 \text{ B} / 1024 = 2,496582 \text{ KB}$. To je 12,85 več kot prvotna slika s 6 indeksiranimi barvami. Če bi povečali dimenzije slike na 120 x 90 slikovnih točk (kar je še zmeraj bolj majhna slika npr. logotip ali gumb), kar je 10 krat po vsaki osi, bi imeli velikost in iste pogoje za prejšnji izračun slike. Dobili bi slikovno datoteko, veliko $11912 \text{ B} / 1024 = 11,63 \text{ KB}$.



Slika 146: Barvna indeksirana slika