

**ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE**

Fakulta elektrotechnická

**Bakalářská práce**

**Květen 2014**

**Tomáš Melechovský**



**ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE**

---

Fakulta elektrotechnická  
Katedra radioelektroniky

**Měření parametrů autostereoskopického displeje**

**Measurement of auto-stereoscopic display parameters**

Bakalářská práce

Studijní program: Komunikace, multimédia a elektronika  
Studijní obor: Multimediální technika

Vedoucí práce: Ing. Karel Fliegel, PhD.

**Tomáš Melechovský**

---

Praha 2014

# Čestné prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

Datum: 23.5.2014

.....

Tomáš Melechovský

České vysoké učení technické v Praze  
Fakulta elektrotechnická

katedra radioelektroniky

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: **Tomáš Melechovský**

Studijní program: Komunikace, multimédia a elektronika  
Obor: Multimediální technika

Název tématu: **Měření parametrů autostereoskopického displeje**

Pokyny pro vypracování:

Popište základní principy zobrazování na autostereoskopickém displeji. Zpracujte přehled metod pro měření důležitých parametrů těchto displejů. Navrhněte metodiku pro měření přeslechu mezi jednotlivými pohledy u autostereoskopického displeje, měření proveďte a analyzujte výsledky.

Seznam odborné literatury:

- [1] Ozaktas, H. M., Onural, L.: Three-Dimensional Television: Capture, Transmission, Display, Springer, 2008.
- [2] Javidi, B., Okano, F.: Three-Dimensional Television, Video, and Display Technologies, Springer, 2002.

Vedoucí: Ing. Karel Fliegel, Ph.D.

Platnost zadání: do konce letního semestru 2014/2015

  
Prof. Ing. Miloš Klíma, CSc.  
vedoucí katedry



  
prof. Ing. Pavel Ripka, CSc.  
děkan

V Praze dne 10. 2. 2014

## **Anotace**

Tato práce se zabývá autostereoskopickými displeji. V úvodu je zaměřena na principy zobrazování na autostereoskopických displejích. Hlavní část práce se zabývá měřením přeslechů nejrozšířenějšího typu těchto displejů, tedy multiview displejů s lentikulárními čočkami, a možnými problémy, znesnadňující provedení tohoto měření, mezi které patří, jako například v našem případě, zjištění prokládacího vzoru displeje. Měření přeslechu je důležitou součástí pro vývoj algoritmů na snížení přeslechu u autostereoskopických displejů.

## **Klíčová slova**

Autostereoskopický displej, multiview displej, měření přeslechu, vzor prokládání

## **Abstract**

This thesis concerns in autostereoscopic displays. Firstly, it's focused on principles of autostereoscopic display. Main part is spent to crosstalk measurement of the most common type of autostereoscopic displays – multiview displays with lenticular lens sheet, and problems that makes it difficult, including the deriving interleaving pattern. Crosstalk measurement has important part in development of crosstalk reduction algorithms for autostereoscopic displays.

## **Index words**

Autostereoscopic display, multiview display, crosstalk measurement, interleaving pattern

# Obsah

Obsah .....	5
1 Úvod.....	7
2 Autostereoskopické displeje .....	8
2.1 Dvoupohledové .....	8
2.2 Vícepohledové .....	9
3 Přeslech.....	11
3.1 Vnější přeslech.....	11
3.2 Vnitřní přeslech.....	11
4 Měření přeslechu.....	13
4.1 Příprava na měření .....	13
4.2 Generování snímků pro měření.....	14
4.3 Skript pro generování testovacích obrázků.....	14
4.4 Měřicí fotoaparát.....	15
5 Zjištění vzoru prokládání displeje.....	16
5.1 Metoda I.....	16
5.2 Metoda II.....	18
5.3 Metoda III .....	19
5.3.1 Odečtení šířky vzoru .....	19
5.3.2 Odečtení výšky vzoru .....	20
5.3.3 Zjištění počtu pohledů .....	21
5.3.4 Sestavení finálních testovacích obrázků .....	21
6 Vlastní měření.....	22
6.1 Zjištění vzoru prokládání .....	22
6.2 Realizace měření přeslechu .....	27
6.3 Vyhodnocení výsledků .....	29
7 Závěr .....	32

8 Seznam použité literatury a zdrojů informací .....	33
9 Přílohy.....	34
9.1 Obsah datového nosiče .....	34
9.2 Vzory prokládání .....	34

# 1 Úvod

Autostereoskopické displeje jsou takové displeje, které dokáží zobrazit 3D obraz tak, že ho uvidíme bez potřeby nošení speciálních brýlí. V běžných obchodech však tyto displeje najedeme pod označením „3D bez brýlí“, v angličtině jako „glasses-free 3D“ či „glassesless 3D“. Jejich princip spočívá v rozdělení obrazu vytvořeného TFT LCD displejem určitou optickou bariérou na několik obrazů, kterým se říká pohledy (views). Optická bariéra může být zkonstruována různými způsoby, jako příklad můžeme uvést třeba paralax bariéru, optické hranoly, lentikulární čočky, různé holografické prvky či zrcadla. Autostereoskopické displeje můžeme tak rozdělit na několik skupin, například na displeje s fixními zónami, kde určité pohledy vidíme v určité zóně, která se vůči displeji nemění, a displeje „head-tracked“ či „eye-tracked“, které sledují polohu pozorovatele (zejména hlavu nebo oči) a přizpůsobují se tak, aby pozorovatel viděl vždy kvalitní 3D obraz bez ohledu na jeho polohu před displejem. Displeje můžeme také rozdělit dle počtu zobrazovaných pohledů, a to na takzvané dvoupohledové „two-view“, tedy displeje, které zobrazují jeden stereo pár pohledů – pro levé a pravé oko. Dalším typem jsou displeje vícepohledové „multiview“ – ty dokáží zobrazit více pohledů najednou, v praxi se běžně setkáváme např. s 7-mi, 9-ti, ale i 28-mi pohledy. Posledním typem v tomto dělení jsou displeje „super multiview“, nazývané také jako „light field“ displeje. Jejich cílem je zobrazit plně kvalitní 3D vjem při pohybu pozorovatele před displejem. K tomu se využívá velkého počtu zobrazovaných pohledů, typicky 64 a více. Konstrukce těchto displejů je velmi náročná a nákladná. [1], [2]

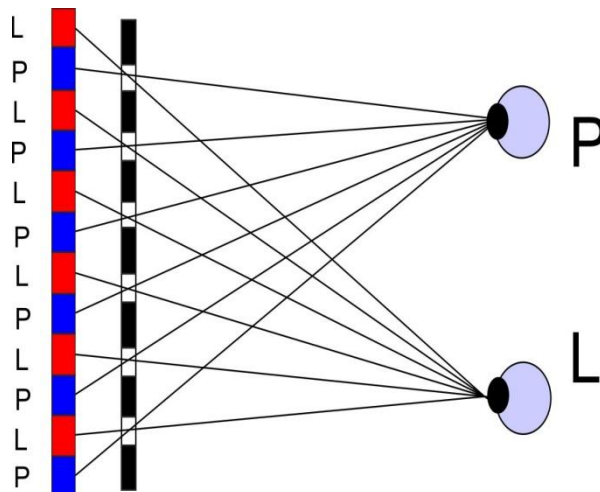


## 2 Autostereoskopické displeje

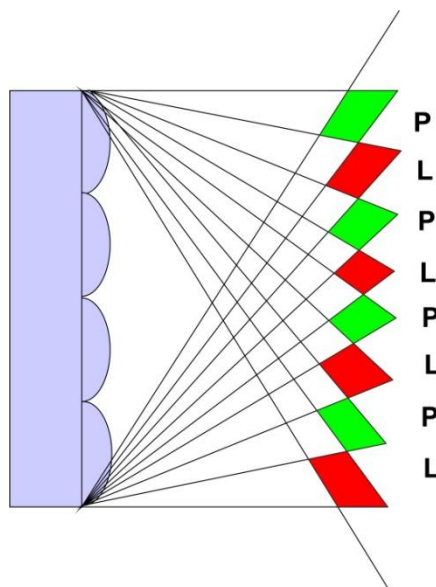
Zaměříme se na displeje s fixními zónami, konkrétně na two-view, ale převážně na multiview displeje.

### 2.1 Dvoupohledové

Dvoupohledový (v angličtině označován jako Two-view) displej zobrazuje, jak již jeho označení napovídá, 2 pohledy, tedy jeden stereo pár. K rozdělení těchto dvou pohledů do určitých oblastí před displejem používá zejména 2 principy, a to konkrétně paralax bariéru a nebo lentikulární čočky. Zatímco princip s paralax bariérou funguje tak, že bariéra blokuje světlo tak, aby bylo vidět pouze v žádoucí oblasti, princip s lentikulárními čočkami využívá refrakce světla, které se rozděluje do požadovaných směrů. Systém s paralax bariérou zpravidla směřuje stereo pár do jednoho místa, tím umožňuje sledovat obraz prakticky jen jednomu pozorovateli, ale lze jej zkonstruovat i tak, že má tento stereo pár „zkopírovaný“ podél horizontální osy displeje. Systém s lentikulárními čočkami zřejmě využívá tento stereo pár zkopírovaný. Kopírování stereo páru podél horizontální osy nám nabízí více pozorovacích zón, na druhou stranu se zde můžeme setkat s pseudoskopickým jevem, který nastává, když obraz pro levé oko vidíme pravým okem a zřejmě i levým okem vidíme obraz pro oko pravé. [2]



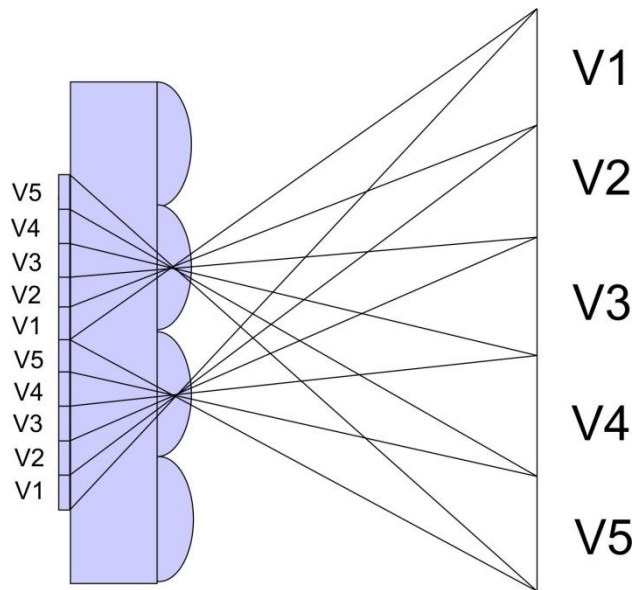
Obr. 1 Princip funkce two-view displeje s paralax bariérou (inspirováno [2])



Obr. 2 Princip funkce two-view displeje s lentikulárními čočkami (inspirováno [2])

## 2.2 Vícepohledové

V dnešní době je však podstatně rozšířenějším typem autostereoskopických displejů vícepohledový displej (v angličtině označovaný jako Multiview). Multiview displeje tedy zobrazují více pohledů, označme si je např.  $v$ . Pak tedy počet stereo párů produkovaných displejem je  $v - 1$ . Právě díky většímu počtu pohledů a tím pádem i stereo párů tyto displeje snižují počet pozic, kde vzniká pseudoskopický jev, a zároveň nabízejí možnost pozorovat obraz z více různých úhlů. Jako optickou bariéru využívají lentikulární čočky, které jsou umístěny na TFT LCD displeji v ohniskové vzdálenosti od samotných pixelů. Ty pak na principu refrakce světla rozdělují obraz z jednotlivých pixelů, přesněji subpixelů, do určených zón příslušných daným pohledům, kde pozorovatel vidí pouze jeden vybraný pohled. Jelikož počet různých pohledů snižuje rozlišení displeje, lentikulární čočky jsou mírně zkoseny tak, aby docházelo ke snížení rozlišení v obou směrech, jak horizontálním tak vertikálním. Díky tomuto zkosení zde však vznikají přeslechy mezi jednotlivými pohledy. Subpixely určené jednotlivým pohledům se střídají tak, aby byly směřovány do pozorovacích zón, ve kterých uvidíme právě daný pohled. Z důvodu zkosení lentikulárních čoček se tyto subpixely musí posouvat tak, aby byly stále ve stejné poloze vůči čočce. Pokud se však podíváme na tyto subpixely, můžeme si všimnout, že se posunem v jednotlivých řádcích nakonec dostanou do rozestavení, ve kterém už jednou byly. Tento blok několika řádků a sloupců nazýváme *vzor prokládání displeje* (viz Obr. 4), který se vlastně opakuje jak v horizontálním, tak vertikálním směru. [1], [2]



Obr. 3 Princip funkce multiview displeje (inspirováno [7])

v	v-1	...	2	1	v	v-1	...	...	...
v-1	...	2	1	v	v-1	...	...	...	...
...	...	...	...	...	...	...	...	...	...
1	v	v-1	...	2	1	v	v-1	...	...
v	v-1	...	2	1	v	v-1	...	...	...

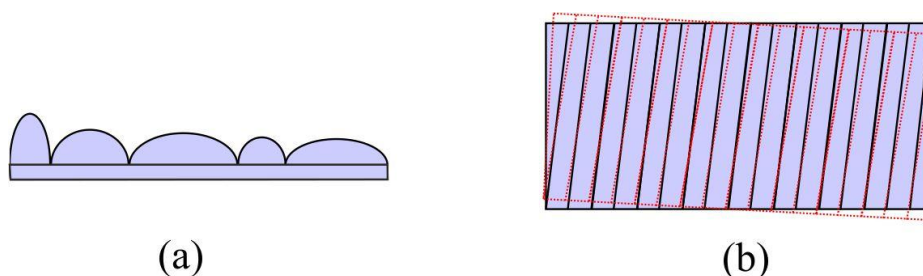
Obr. 4 Příklad vzoru prokládání

## 3 Přeslech

Přeslech je obecně nežádoucí signál, který přijímáme spolu s žádoucím signálem v jednom určitém kanálu. Zde si jako kanál můžeme představit danou pozorovací zónu, kde žádoucí signál je očekávaný obraz příslušící jistému pohledu a nežádoucí signál je zřejmě obraz příslušný pohledům okolním. Přeslech u autostereoskopických displejů můžeme dle původu rozdělit na dva typy, na vnější a na vnitřní přeslech. Toto dělení je podstatné zejména při tvorbě algoritmů, které úpravou obrazu sníží přeslech, samotný vliv obou typů přeslechu je podobný. Kvůli přeslechu vznikají v obraze různé artefakty, mezi nejčastější patří efekt dvojitých hran, neboli tzv. „duchové“. Právě tyto artefakty nám při sledování 3D obrazu narušují, a v extrémním případě i znemožní, kvalitní 3D vjem. Největší přeslech vzniká mezi sousedními pohledy, jistý příspěvek však mají i pohledy další. [3], [4], [5], [6]

### 3.1 Vnější přeslech

Vnější přeslech je způsoben chybami při výrobě lentikulárních čoček displeje, případně při konstrukci displeje. Mezi tyto chyby patří např. nestejnost čoček, nehomogenita ohniskové vzdálenosti a různé lokální deformace (viz Obr. 4 (a)). Jako konstrukční chyby řadíme např. nepřesnost umístění rastru čoček na displej, jak posuvem, tak natočením (viz Obr. 4 (b)). Právě na chybu v natočení je displej asi nejcitlivější. [6]

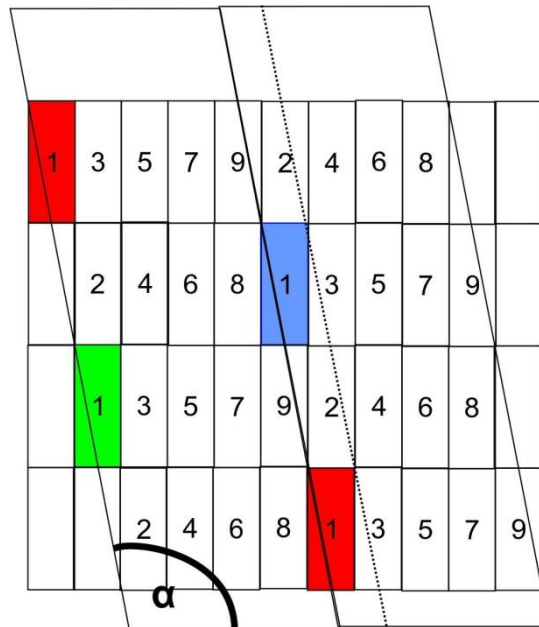


Obr. 4 Příklady vnějšího přeslechu (a) deformace čoček, (b) nepřesné umístění  
(inspirováno [6])

### 3.2 Vnitřní přeslech

Příčinou vnitřního přeslechu displeje je prakticky jeho vlastní princip a proto je nevyhnutelný i v případě, že bychom měli ideální displej bez jediné výrobní chyby, jelikož vnitřní přeslech vzniká díky zkosení lentikulárních čoček na rastru tak, aby docházelo ke snížení rozlišení v obou směrech displeje. Čočka zde překrývá pouze větší část subpixelů náležícím jednomu pohledu, ale i část subpixelů pohledů sousedních. Jak můžeme vidět na

Obr. 5, čočka, která směřuje pohledu č.1 do jednoho směru, částečně prochází přes pohled č.9 a také, pokud si představíme rovnoběžnou čáru procházející druhým rohem subpixelu, pohled č.3. Tento přeslech by byl teoreticky konstantní, pokud bychom měli ideální displej bez výrobních chyb, které nám způsobují vnější přeslechy. [5], [6]

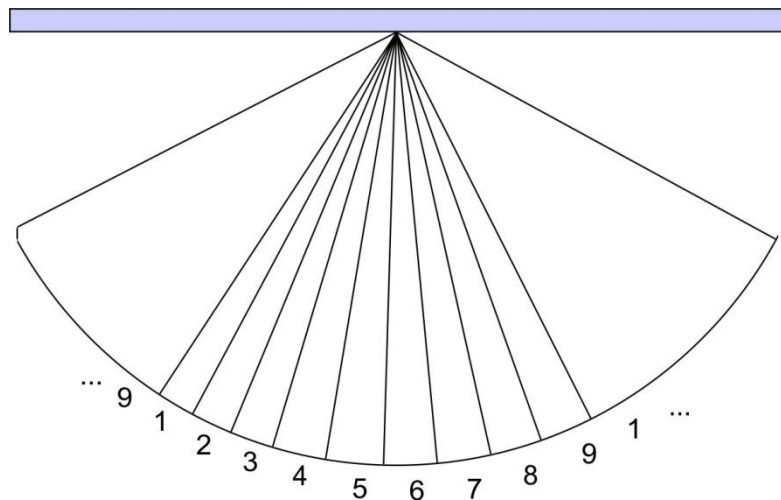


Obr. 5 Příklad struktury multiview displeje (inspirováno [2])

## 4 Měření přeslechu

### 4.1 Příprava na měření

Pro měření přeslechu použijeme fotoaparát a budeme vyhodnocovat jas příslušných snímků, které pořídíme z jednotlivých měřících pozic a které budou příslušet různým pohledům. Jak můžeme vidět na Obr. 5, každému pohledu odpovídají příslušné subpixely. Ty se budeme snažit vybudit příslušnými obrázky tak, abychom rozsvítili právě ty subpixely, které náleží jednomu pohledu.



Obr. 6 Příklad rozložení pohledů v prostoru před displejem (inspirováno [4])

Na Obr. 6 vidíme, jak se pohledy střídají podél horizontální osy displeje. Pro měření si vyznačíme měřící body podél displeje v ideální pozorovací vzdálenosti. Pro každý pohled si vyznačíme místo uprostřed pohledu a dále další body (alespoň 3) mezi středy. Před zahájením měření místnost zatemníme, v případě odrazu světla od podlahy a dalších ploch v blízkosti displeje pokryjeme tyto místa nejlépe černou látkou. Měření začneme umístěním stativu přesně na předem určené místo (označíme ho např. měřící bod č.1) a postupně budeme vybuzovat skupiny pixelů příslušné jednotlivým pohledům (např. pixely s č.1 na Obr. 5, zatímco ostatní budou nerozsvícené, poté s č.2 atd.) a následně snímat. Také vyfotíme displej se všemi pixely rozsvícenými a se všemi pixely zhasnutými, abychom mohli provést kalibraci kvůli odstranění přidaného světla. Po vyfocení všech pohledů se posuneme na další měřící bod a postup opakujeme, až se nám podaří změřit všechny pohledy ve všech vyznačených místech. Pro vyhodnocení využijeme skript, který snímek kalibruje, dále najde na snímku displej a ořízne okraje tak, abychom měli pouze

obrázek displeje. Dále spočítá průměrnou hodnotu jasu daného snímku a vykreslí graf. Po vyhodnocení tak získáme jednotlivé příspěvky pohledů pro danou pozorovací pozici. [4]

## 4.2 Generování snímků pro měření

Pro měření budeme potřebovat  $v+2$  různých snímků, pro vybuzení jednotlivých pohledů a další dva pro kalibraci. Všechny snímky budou prakticky černobílé, jelikož bílá barva se skládá ze všech složek RGB, tudíž získáme největší jas. Ve skutečnosti ale budou tvořeny červenými, modrými a zelenými body, protože jednomu pohledu vlastně přísluší jen část pixelu, tedy jeden subpixel. Pokud se však budeme dívat na displej z dostatečné vzdálenosti, tak nám, i díky lentikulárním čočkám, jednotlivé subpixely splynou. Největším problémem je skutečnost, že neznáme vzor prokládání displeje, tedy nevíme, který subpixel patří tomu kterému pohledu. To znamená, že budeme muset zjistit tento vzor, abychom poté mohli přesně vygenerovat potřebné snímky. K tomu použijeme různé pruhy, které budeme posouvat po obrazovce a sledovat, jak se budou promítat. Způsoby, kterými toho lze dosáhnout, najdeme v další kapitole.

Samotné snímky, které budeme měřit, budou tvořeny černými a RGB body tak, že základem bude černá plocha a RGB body budou přesně v místech subpixelů pro jeden příslušný pohled. Celý snímek pak bude vypadat jako černá plocha s diagonálními proužky, které svým sklonem budou odpovídat sklonu lentikulárních čoček na obrazovce displeje.

## 4.3 Skript pro generování testovacích obrázků

Pro generování obrázků používáme skript v programu MATLAB. Generování probíhá ve dvou krocích. Prvním je vytvoření matice o velikosti  $3X \times Y$ , kde  $X$  je šířka obrázku v pixelech (odpovídá maximálnímu rozlišení displeje) a  $Y$  je výška obrázku v pixelech. Šířku počítáme třikrát, jelikož nás zajímají přímo subpixely displeje. Do této matice zapisujeme, např. hodnotami 0 a 1, které subpixely chceme rozsvítit. V druhém kroku tuto matici přepíšeme do výstupní matice o velikosti  $X \times Y$ , kam pomocí podmínky, využívající např. modulo 3, a definované palety barev (RGB) zapíšeme barvu pixelu, která vlastně odpovídá subpixelu, jenž chceme rozsvítit. Výsledný obrázek uložíme v nekomprimovaném formátu, např. BMP, TIFF.

#### **4.4 Měřicí fotoaparát**

K měření použijeme fotoaparát, který musíme nastavit dle následujících pokynů. Citlivost fotoaparátu (ISO) nastavíme na nejnižší hodnotu. Dále nastavíme nejmenší aperturu – tomu odpovídá největší clonové číslo. Tímto nastavením omezíme šum CCD snímače fotoaparátu na minimum. Posledním parametrem je expoziční doba, tu nastavíme experimentálně tak, aby při maximálním jasu (rozsvítíme všechny subpixely na displeji – tedy celou obrazovku bílou barvou) nebyl snímač fotoaparátu v saturaci. To poznáme jednoduchou kontrolou po zachycení snímku na displeji fotoaparátu např. pomocí zebra efektu nebo podobných funkcí, kde se zobrazí saturovaná oblast. S tímto nastavením využijeme maximální rozsah fotoaparátu pro měření jasu. Vzhledem k tomu, že nás zajímají výstupní snímky jako surová data, odpadá nám nastavení vyvážení bílé barvy. [1]



## 5 Zjištění vzoru prokládání displeje

K zjištění vzoru prokládání jsme si připravili rovnou několik metod, některé jsou poměrně jednoduché, některé složitější. Důvodem je skutečnost, že se nám pomocí dané metody nemusí povést vzor odečíst, jelikož displej dle dokumentace nabízí podstatně větší množství pohledů, než tomu bylo ve studovaných materiálech, a tedy může používat rafinovanější a sofistikovanější vzor prokládání.

### 5.1 Metoda I

Pro zjištění vzoru prokládání dle první metody budeme předpokládat zřejmou strukturu (viz Obr. 5) [5], kterou si ověříme. Předpokládáme tedy, že subpixely jednotlivých pohledů jdou rovnoběžně s lentikulárními čočkami tak, aby se všechny lámaly do jednoho směru. Základním parametrem je zřejmě úhel sklonu lentikulárních čoček  $\alpha$  (viz Obr. 4). Pomocí něj pak zjistíme směrnici přímky, která vytváří vybuzený pruh. Úpravou směrnice rovnice přímky (1) pak získáme vzorec pro výpočet souřadnice  $x$  (2) :

$$y = kx + q, k = \tan \alpha \quad (1)$$

$$x = \frac{y}{k} - \frac{q}{k} \quad (2)$$

Může zde nastat několik možných výsledků. V ideálním případě tato struktura bude odpovídat, navíc však zjistíme, jestli naším vygenerovaným vzorem můžeme budít přímo jednotlivé subpixely, to znamená, že pokud bychom chtěli pro pohled č.2 nastavit bílou barvu, museli bychom vybudít subpixely (viz Obr. 5(a)). Další možností je, že zjistíme, že můžeme vybudít pouze celé pixely, tedy celou skupinu RGB, to by například znamenalo, že nastavením bílé barvy v prvním řádku na Obr. 5(a) vznikla v pohledu č. 3 červená, v pohledu č.2 zelená a v pohledu č.1 modrá barva (viz Obr. 6), v tom případě bychom pro pohled č.2 museli vybudít pixel v prvním řádku zelenou a v druhém modrou barvou, ve třetím řádku bychom museli vybudít následující pixel červenou barvou. Na Obr. 5(b) můžeme vidět jednu z dalších variant vzorů. Zde jsou seskupeny pohledy s lichými a sudými čísly k sobě a střídají se v blocích jak v řádcích, tak i ve sloupcích pod čočkami. Zde můžeme vidět, že v prvním řádku jsou nejprve liché pohledy a následují je sudé, v dalším řádku jsou ve stejném sloupci sudé pohledy, které jsou z obou stran obklopeny lichými. Podobných variant může být více, některé např. nestřídají pohledy ve sloupcích, takže v prvním řádku by byly pouze liché pohledy, některé třeba seskupují pohledy ob 2, tedy např. 2, 5, 8, ...

R	G	B	R	G	B	R	G	B	R	R	G	B	R	G	B	R	G	B	R
v	v-1	...	2	1	v	v-1	...	2	1	v	v-2	...	3	1	v-1	v-3	...	4	2
1	v	v-1	...	2	1	v	v-1	...	2	1	v-1	v-3	...	4	2	v	v-2	...	3
2	1	v	v-1	...	2	1	v	v-1	...	4	2	v	v-2	...	3	1	v-1	v-3	...
...	2	1	v	v-1	...	2	1	v	v-1	...	3	1	v-1	v-3	...	4	2	v	v-2

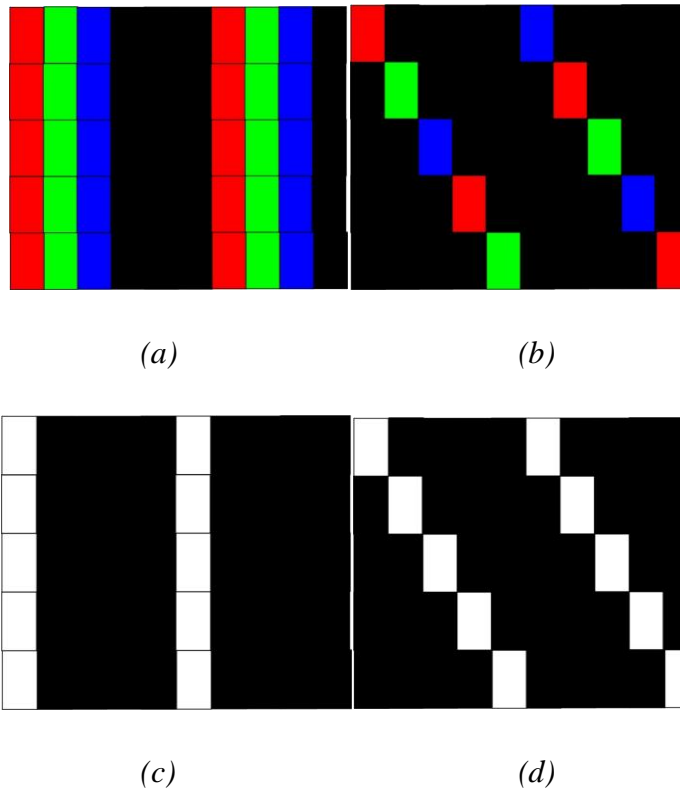
(a) (b)

Obr.5 Předpokládaná struktura displeje (a) (inspirováno [5]), (b)varianta, kde jsou liché a sudé pohledy seskupeny v blocích a střídají se, příklad pro  $v =$  liché číslo

R	G	B	R	G	B	R	G	B	R
v	v-1	...	2	1	v	v-1	...	2	1
1	v	v-1	...	2	1	v	v-1	...	2
2	1	v	v-1	...	2	1	v	v-1	...
...	2	1	v	v-1	...	2	1	v	v-1

Obr. 6 Výsledek pro vybuzení bílou barvou, pokud bych mohli vybudit pouze celý pixel

Pro buzení displeje budeme tedy využívat tyto vzory: 3 svislé pruhy barev RGB a zbytek černá barva (pro sadu 28 pruhů), šikmý pruh barev RGB (viz Obr. 5), svislý pruh bílé barvy (zbytek černý, opět pro jednu sadu), šikmý bílý pruh (podobně jako pro barevný).



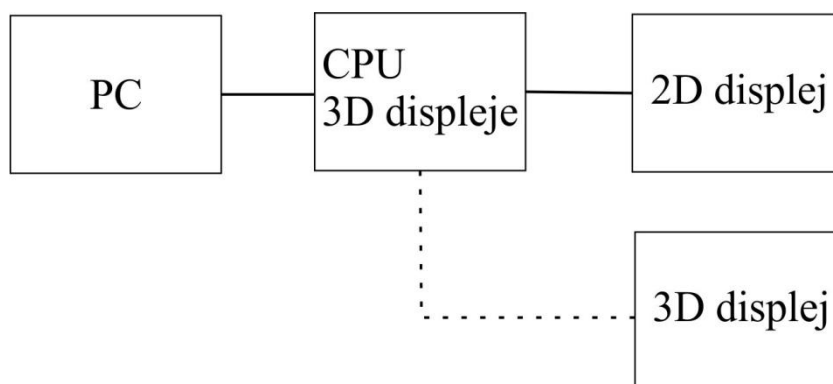
*Obr. 7 Vzory pro buzení displeje, (a) svislé RGB pruhy, (b) šikmé RGB pruhy, (c) svislé bílé pruhy, (d) šikmé bílé pruhy*

Jednou z možných variant je také to, že se náš předpoklad nepotvrdí. Poté v lepší situaci se nám i tak podaří zjistit strukturu displeje, nebo alespoň zjistíme, jak bude možné displej budít. Pokud by se nám nepodařilo zjistit nic, budeme muset použít jinou metodu.

## 5.2 Metoda II

Další metodou je připojení výstupu z počítače vzadu na displeji na obyčejný 2D displej, přičemž vstupním snímkem bude 3D obrázek (viz Obr.8). Na 2D displeji by se tak měl objevit obraz, který bude mít jednotlivé pixely umístěny tak, že budou přímo odpovídat struktuře displeje. Zde může nastat problém, že počítač nebude fungovat, pokud ho odpojíme od našeho 3D displeje. Tuto metodu lze upravit tak, že k připojenému 2D displeji paralelně připojíme zpět náš 3D displej (viz Obr.8). Tato „upravená“ varianta bude fungovat pouze pro náš displej, případně pro displeje spolupracující s Dimenco softwarem. Poté použijeme program Dimenco Control Tool, který nabízí různé možnosti nastavení. Zde využijeme možnost natáčení kuželu pozorovacího pohledu. To by nám mělo způsobit, že se pixely příslušící jednotlivým pohledům „posunou“ tak, aby odpovídaly daným pohledům pro takto posunutý kužel. Toto nastavení najdeme v programu Dimenco Control

Tool v kartě Advanced pod názvem Viewing Direction. Základní nastavení je na hodnotě 0. Dále program umožňuje například měnit ideální pozorovací vzdálenost, tu však pro toto měření měnit nebudeme. [9]



Obr. 8 Schéma zapojení pro „druhou metodu“ (bez 3D displeje). Pro zapojení „upravené metody“ připojíme i kabel k 3D displeji (čárkovaně)

### 5.3 Metoda III

Poslední metoda spočívá v měření úhlové viditelnosti zobrazovaných subpixelů, přičemž jako měřicí prostředek budeme využívat fotoaparát. Jím pořizujeme snímky ze vzdálenosti nižší, než je optimální pozorovací vzdálenost (dále jako *snímky zblízka*). Právě snímek zblízka nám umožňuje zachytit a následně i vyhodnotit, jak bychom viděli rozsvícený subpixel z několik úhlů zároveň. Tato metoda se skládá ze 4 částí, konkrétně z odečtení šířky vzoru, odečtení výšky vzoru, zjištění skutečného počtu pohledů a finálního sestavení vzoru prokládání. Tato metoda je zdaleka nejsložitější, avšak slibuje velmi nadějně výsledky. K její realizaci budeme potřebovat fotoaparát, který si připravíme dle kapitoly 4.4 Měřicí fotoaparát. Snímky budeme pořizovat z co nejbližší vzdálenosti, v našem případě je to vzdálenost okolo 20 centimetrů.

#### 5.3.1 Odečtení šířky vzoru

Prvně si vytvoříme testovací obrázek s čtvercem o známé velikosti pixelů. Ten zobrazíme na displeji a uděláme první snímek zblízka. Je vhodné, aby byl čtverec na snímku celý. Nyní zjistíme rozměry čtverce na snímku v pixelech a vypočítáme si konstantu  $k$ , která představuje poměr mezi rozměry čtverce na snímku zblízka a rozměry čtverce na testovacím obrázku.

Dalším krokem je vytvoření  $n$  testovacích obrázků, pomocí kterých zjistíme šířku vzoru. Na každém obrázku bude několik řádků, v každém rozsvítíme subpixely s určitým rozestupem. Je vhodné nechat mezi rozsvícenými řádky několik řádků zatemněných (černých). Na každém obrázku tedy bude odstup mezi subpixely  $n$ , který se bude pro každý obrázek zvyšovat o 1, v celkovém intervalu např.  $\langle 1;100 \rangle$ , maximální hodnotu intervalu si zvolíme přiměřeně vzhledem k očekávanému počtu pohledů, např. na hodnotou  $2 \cdot v$ . Tento obrázek zobrazíme na displeji a pořídíme snímek zblízka.

Tyto snímky pak vyhodnotíme. Musíme rozlišit, kde je v daném řádku mezera vzniklá rozestupem subpixelů a kde je mezera blokována optickou bariérou. K tomu nám pomůže filtr maximálních hodnot, který aplikujeme na snímky. Rozměr okna nastavíme pomocí vztahu

$$okno_{1D} = \left( n \cdot \frac{k}{3} \right) \times 1, \quad (3)$$

kde  $\frac{k}{3}$  respektuje velikost subpixelu vůči celému pixelu. Po projití snímku filtrem vyhodnotíme *oddělení viditelnosti* jednotlivých řádků, např. pomocí poměru

$$OV_r = \frac{N}{V}, \quad (4)$$

kde  $N$  je „neviditelná“ část řádku a  $V$  je „viditelná“ část řádku. Subpixely náležící jednomu pohledu zde budou mít nejvyšší poměr  $OV_r$ , jelikož jejich úhlová viditelnost je teoreticky stejná. Jako hodnotu  $n$  se snažíme vybrat co nejmenší, ale zároveň s co možná nejvyšším poměrem  $OV_r$ . [1]

### 5.3.2 Odečtení výšky vzoru

Když jsme zjistili šířku vzoru, je na řadě zjištění jeho výšky. K tomu použijeme testovací obrázky, kde jednotlivé řádky budou mít změřený odstup mezi subpixely  $n$ , ale budeme měnit rozestup mezi jednotlivými řádky  $m$  ve vertikálním směru. Pro každý obrázek pořídíme snímek zblízka a na každý snímek aplikujeme filtr maximálních hodnot s dvourozměrným oknem o velikosti vypočtené dle vztahu :

$$okno_{2D} = \left( n \cdot \frac{k}{3} \right) \times (m \cdot k) \quad (5)$$

S měnícím se  $m$  nám budou čáry na snímku rotovat, což je způsobeno měnící se úhlovou viditelností jednotlivých subpixelů. Hledanou hodnotu  $m$  najdeme podle snímku, ve kterém

zjistíme nejnižší frekvenci ve vertikálním směru, což znamená, že právě tyto subpixely mají stejnou úhlovou viditelnost. To se dá zjednodušeně označit jako nejmenší zkosení jednotlivých čar. [1]

### 5.3.3 Zjištění počtu pohledů

Zjistili jsme už rozměry vzoru prokládání  $n \times m$ , čemuž by teoreticky odpovídal maximální počet různých pohledů  $n*m$ . Většinou je však počet pohledů menší, neboť subpixely s různými souřadnicemi patří jednomu samému pohledu. I přesto, že v našem případě výrobce udává počet pohledů displeje, může se stát, že byl v tomto údaji příliš optimistický, a údaj jím zadaný nebude odpovídat skutečnosti.

Vytvoříme si tedy  $n*m$  různých testovacích obrázků, kde rozsvítíme vždy subpixel, který má stejné souřadnice v rámci vzoru prokládání. Pro každý obrázek uděláme snímek zblízka. Na každý snímek opět aplikujeme filtr maximálních hodnot s dvourozměrným oknem  $(n*k/3) \times (m*k)$ . Testovací obrázky s rozsvícenými subpixely, které přísluší jednomu samému pohledu, budou mít velmi podobnou úhlovou viditelnost. To znamená, že jejich snímky se budou minimálně lišit, zatímco snímky jiných pohledů se budou lišit znatelně více. K rozpoznání těchto rozdílů uděláme normalizovaný rozdíl snímků  $l^2$  a z jeho hodnot sestavíme histogram. [1]

### 5.3.4 Sestavení finálních testovacích obrázků

Pokud jsme si určili a našli ve vzoru prokládání subpixely náležící jednomu pohledu, pak sestavení testovacího obrázku pro daný pohled bude už jen maličností. Vygenerujeme si tedy obrázky, které mají rozsvícené všechny subpixely, jež patří jednomu samému pohledu. V rámci jednoho vzoru prokládání jich bude více s různými souřadnicemi. Jednotlivé pohledy si označíme např. číslicemi, nejlépe tak, jak se zobrazují v prostoru, např. zleva do prava. [1]

## 6 Vlastní měření

V této části se zaměřím na popis vlastního měření mnou provedeného. Měření jsem prováděl na displeji Philips BDL4251VS. Jeho parametry jsou uvedeny v následující tabulce.

Philips BDL4251VS	
Úhlopříčka	42" / 107 cm
Rozlišení displeje	1 920 × 1 080
Pozorovací úhel	150°
Počet pohledů	28
Jas	700 cd/m <sup>2</sup>
Kontrast	1 100:1
Doba odezvy	9 ms

Tab. 1 Vybrané parametry displeje Philips BDL4251VS [8]

### 6.1 Zjištění vzoru prokládání

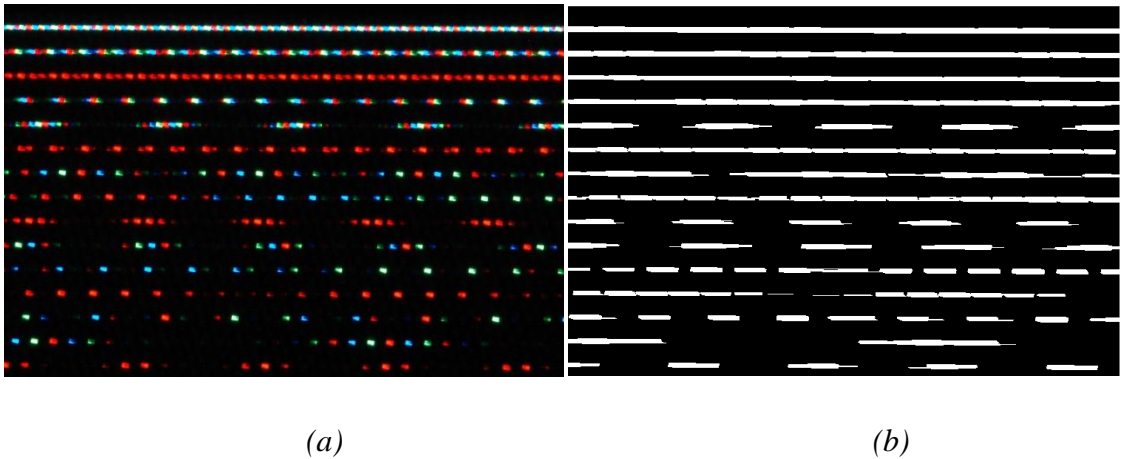
V prvním kroku jsem si vygeneroval testovací obrázek pomocí metody I (viz kapitola 5.1). Prvním nastaveným úhlem sklonu odhadovaným na  $\alpha = 75^\circ$ . Tento obrázek jsem si promítl na displeji a hned na první pohled bylo patrné, že bude potřeba úhel sklonu zvětšit. Úhel jsem tedy zvětšoval tak, aby na displeji nebyly vidět jednotlivé pruhy, ale barva se „slila“ tak, jakoby svítil celý displej. Tímto způsobem jsem určil úhel sklonu  $\alpha = 80,61^\circ$ . Dalším krokem bylo určení rozestupu jednotlivých subpixelů tvořících diagonální pruhy. Tuto hodnotu jsem určil na základě vygenerování testovacích obrázků s různým rozstupem mezi rozsvícenými pruhy. Nejlepších výsledků jsem dosáhl při rozstupu o 14 subpixelů. Posunem těchto pruhů po horizontální ose jsem realizoval buzení různých pohledů. Problémem zde bylo, že výsledný obraz na displeji se několikrát zatemnil a několikrát byl plně rozsvícen, což neodpovídalo teoretickému předpokladu. Z důvodu několika zlomů v obrázku jsem vyzkoušel změnit úhel sklonu  $\alpha = 80,538^\circ$ , což odpovídá sklonu pro  $\arctan(6)$ , kde 6 je poměr vybuzených subpixelů, resp. pixelů – přímka přechází přes 1 pixel na šířku a 6 pixelů na výšku. Tyto zlomy zmizely, ale na displeji jsem již nemohl pozorovat zatemňování a následné rozsvícení, místo toho zde byly dva až tři silné pruhy, které se se změnou pozice vůči displeji pohybovaly. Z tohoto důvodu jsem zvolil k zjištění vzoru prokládání metodu III.

Pixel 1			Pixel 2			Pixel 3			Pixel 4			Pixel 5	
R	G	B	R	G	B	R	G	B	R	G	B	R	G
1	3	5	7	9	11	13	15	17	2	4	6	8	10
12	14	16	18	1	3	5	7	9	11	13	15	17	2
4	6	8	10	12	14	16	18	1	3	5	7	9	11
13	15	17	2	4	6	8	10	12	14	16	18	1	3
5	7	9	11	13	15	17	2	4	6	8	10	12	14
16	18	1	3	5	7	9	11	13	15	17	2	4	6
8	10	12	14	16	18	1	3	5	7	9	11	13	15
17	2	4	6	8	10	12	14	16	18	1	3	5	7
9	11	13	15	17	2	4	6	8	10	12	14	16	18
1	3	5	7	9	11	13	15	17	2	4	6	8	10
12	14	16	18	1	3	5	7	9	11	13	15	17	2
4	6	8	10	12	14	16	18	1	3	5	7	9	11
13	15	17	2	4	6	8	10	12	14	16	18	1	3
5	7	9	11	13	15	17	2	4	6	8	10	12	14
16	18	1	3	5	7	9	11	13	15	17	2	4	6
8	10	12	14	16	18	1	3	5	7	9	11	13	15
17	2	4	6	8	10	12	14	16	18	1	3	5	7
9	11	13	15	17	2	4	6	8	10	12	14	16	18
1	3	5	7	9	11	13	15	17	2	4	6	8	10

Tab. 2 Vzor prokládání zjištěný metodou III

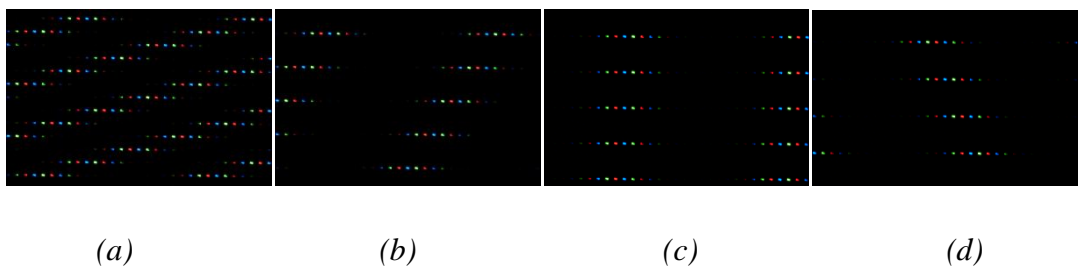
Nejdříve jsem si připravil testovací obrázky s řádky, ve kterých byly rozsvíceny subpixely s různým rozestupem. Tyto obrázky jsem si postupně promítal na displeji a pořídil jejich snímky zblízka, jež jsem následně vyhodnotil. K tomu jsem použil filtr maximálních hodnot s rozměrem okna  $1 \times (n*4)$ , kde 4 odpovídá přibližně  $k/3$ . Konstantu  $k$  jsem mezitím vypočítal ze snímku čtverce o rozměru  $10 \times 10$  pixelů, který měl na snímku zblízka rozměr  $110 \times 106$  pixelů. Na snímcích, které prošly filtrem, jsem hledal takový, který bude mít větší oddělení viditelnosti oproti ostatním snímkům, ale zároveň musí být jeho  $n$ , tedy rozestup mezi subpixely, co nejnižší (viz Obr. 9). V mém případě  $n = 14$ , jak můžeme vidět na Obr. 9(b), oddělení viditelnosti řádku č.14 (druhý zespoda) je největší, čehož si všimneme i pouhým okem.



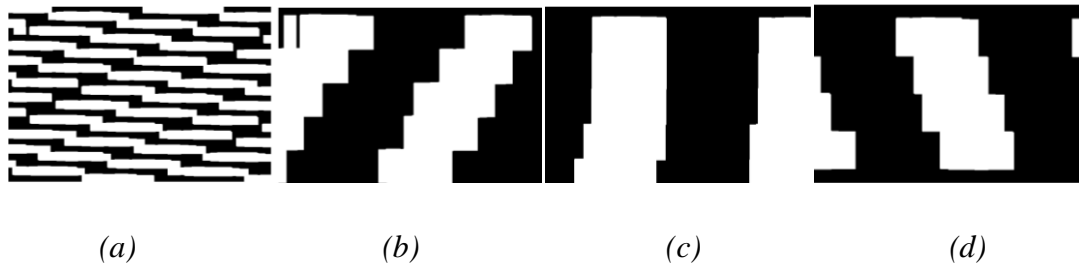


Obr. 9 Snímky z měření šířky vzoru – ukázka více rozestupů (a) snímek displeje, (b) snímek po průchodu filtrem

Dále je potřeba odečíst výšku vzoru. K tomu jsem si vygeneroval snímky s rozestupem mezi subpixely  $n=14$  a s různým rozestupem mezi celými řádky  $m$ . Tyto snímky jsem opět vyfotil zblízka a provedl jejich analýzu. Zde jsem použil filtr maximálních hodnot s dvourozměrným oknem  $(m*10) \times 56$  a hodnotu  $m$  jsem hledal takovou, kde měl snímek nejmenší vertikální frekvenci, opět s co nejmenším rozestupem  $m$ . Na následujících obrázcích můžeme pozorovat rotaci zmíněnou v teoretické části (kapitola 5.3.2). Nejmenší vertikální frekvenci můžeme sledovat na snímku pro řádkování  $m = 19$  (viz Obr. 10(c)). Ještě patrnější rozdíl vidíme na snímcích po průchodu filtrem maximálních hodnot (viz Obr. 11).

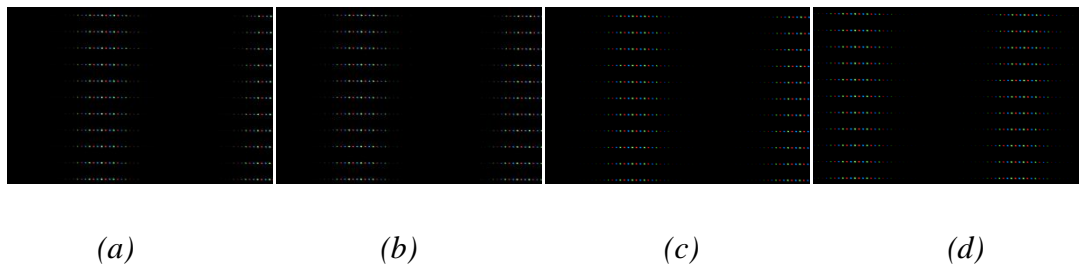


Obr. 10 Snímky displeje z měření výšky vzoru (a)  $14 \times 7$ , (b)  $14 \times 18$ , (c)  $14 \times 19$ , (d)  $14 \times 20$



Obr. 11 Snímky z měření výšky vzoru po průchodu filtrem (a)  $14 \times 7$ , (b)  $14 \times 18$ , (c)  $14 \times 19$ , (d)  $14 \times 20$

Poté jsem si vygeneroval obrázky se stejnými rozsvícenými subpixely tak, aby respektovaly změřené rozměry vzoru. Celkový počet těchto obrázků je 266 ( $19 \cdot 14$ ) a pro přehlednost jsem je označil dvojčíslicím, kde první číslo značí řádek a druhé číslo sloupec, tedy např. snímek 2\_5 má rozsvícený subpixel v 2. řádku a 5. sloupci v rámci vzoru prokládání. Ke všem těmto obrázkům jsem pořídil snímky zblízka (viz Obr. 12), aplikoval na ně stejný filtr maximálních hodnot s oknem o velikosti  $190 \times 56$  (viz Obr. 13) a vypočítal normalizovaný rozdíl mezi jednotlivými snímky  $l^2$ , který jsem získal odečtením dvou snímků od sebe a poté kvadrátem tohoto rozdílu (viz Obr. 14). Ten jsem vynesl do histogramu a určil hranici, pod kterou patří snímky s minimálním rozdílem v úhlové viditelnosti a tedy i snímky, které patří stejnému pohledu.



Obr. 12 Snímky rozsvícených jednotlivých subpixelů ve vzoru prokládání (a) subpixel 1\_1, (b) subpixel 1\_10, (c) subpixel 2\_5, (d) subpixel 19\_14



Obr. 13 Snímky rozsvícených jednotlivých subpixelů po průchodu filtrem (a) subpixel 1\_1, (b) subpixel 1\_10, (c) subpixel 2\_5, (d) subpixel 19\_14

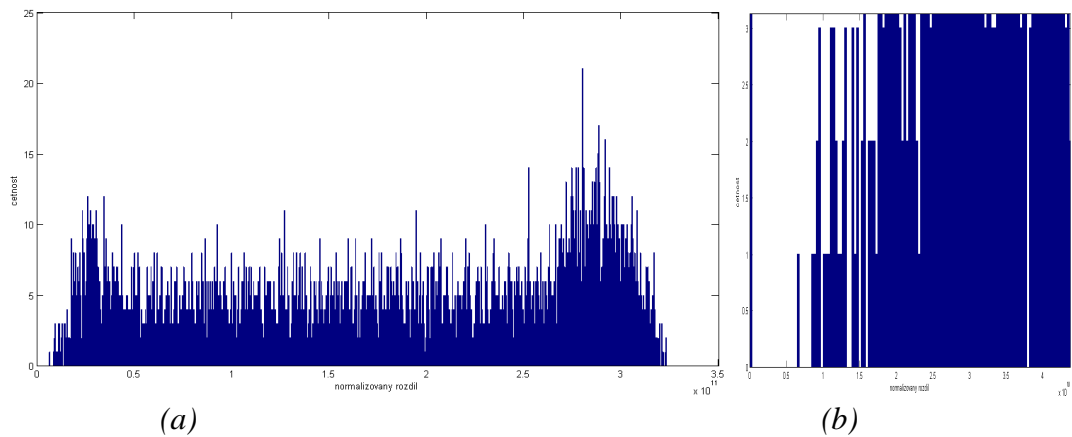


(a)

(b)

(c)

Obr. 14 Snímky normalizovaného rozdílu pro porovnávání jednotlivých subpixelů  
 (a) rozdíl 1\_1 – 1\_10, (b) rozdíl 1\_1 – 2\_5, (c) rozdíl 1\_1 – 19\_14



(a)

(b)

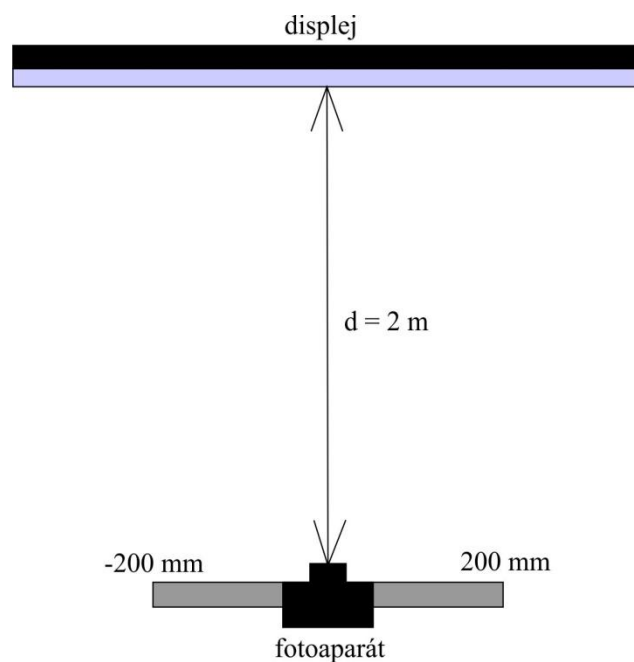
Obr. 15 Histogram normalizovaných rozdílů pro 18 snímků porovnaných se všemi 266 (a) celý histogram, (b) zvětšení pro nás zajímavé části

Histogram (viz Obr. 15) zobrazuje četnost normalizovaných rozdílů mezi jednotlivými snímky, zde jsem vytvořil histogram, ve kterém jsou rozdíly prvních 18 snímků. Každý snímek byl porovnán se všemi 266 snímky. To hlavně z důvodu výpočetní náročnosti tohoto skriptu, jelikož jsem ho musel použít dvakrát, poprvé pro získání hranice a podruhé pro celý výpočet. Dalším důvodem je čitelnost grafu, protože při kompletním provedení by zde byl velký počet hodnot a hranice by se špatně určovala. Takto předpokládám, že další hodnoty budou velmi podobné, případná chyba by v dalším kroku zřejmě byla odhalena a opravena. Na Obr. 15(a) tedy vidíme celý histogram, nicméně nás zajímá část s nejmenšími rozdíly, abychom určili hranici, pod kterou budou právě rozsvícené subpixely patřit jednomu pohledu. Tato část je zvětšena na Obr. 15(b), kde můžeme vidět mezeru v oblasti  $1,3 - 1,4 \cdot 10^{11}$ , v této oblasti jsem určil rozhodovací úroveň. Podle tohoto vyhodnocení však odpovídal maximální počet pohledů  $\nu = 9$ . Dle pořadí snímků a mnou získaných dat jsem následně sestavil vzor prokládání. Ten nebyl úplně dokonalý a jeho promítnutí na displeji se vůbec nechovalo tak, jak bych očekával. Upravil jsem tedy počet pohledů na  $\nu = 18$ , jelikož snímky se sudými násobky původních označení

pohledů měly svou úhlovou viditelnost mírně posunutou, a seřadil jsem liché a sudé pohledy za sebe tak, jak se měnila úhlová viditelnost, tedy pokud snímek 1\_10 byl podobný snímku 1\_1, ale posunutý směrem k snímku 1\_2, označil jsem ho jako pohled č.2, zatímco snímek 1\_1 patřil pohledu č.1 a snímek 1\_2 pohledu č.3. Takto upravený vzor už na displeji vybudil měnící se obraz. Paradoxně se však velmi podobal obrazu, který jsem získal pomocí metody I při úhlu sklonu  $\alpha = 80,538^\circ$ . Největším problémem u této metody byla potřeba velkého výpočetního výkonu. Z tohoto důvodu nebylo možné provést měření tolikrát, abych získal optimální výsledky.

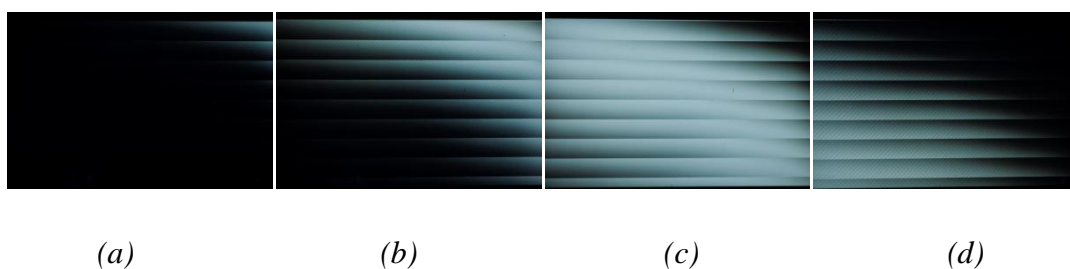
## 6.2 Realizace měření přeslechu

K měření přeslechu jsem si připravil fotoaparát Nikon D70 dle kapitoly 4.4. Jeho rozlišení je  $3\,008 \times 2\,000$  pixelů a výstupní formát surových dat je NEF. Pro měření jsem fotoaparát používal v režimu M, tedy s manuálním nastavením, kde můžeme měnit libovolně parametry fotoaparátu. Citlivost snímače ISO lze nastavit buď v menu fotoaparátu, nebo po stisku tlačítka ISO otáčením hlavním ovládacím kolečkem. Hodnotu ISO jsem nastavil na nejnižší hodnotu 200. K nastavení apertury slouží v manuálním režimu pomocné ovládací kolečko, nejmenší apertuře odpovídá největší clonové číslo, zde je tato hodnota udávána jako „f/-number“. S použitím makroobjektivu jsem nastavil tuto hodnotu na f/44. Expoziční dobu je označena jako „shutter speed“ a nastavuje se v manuálním režimu hlavním ovládacím kolečkem. Tuto hodnotu jsem nastavoval při plně vybuzeném displeji a optimální doba byla 1,3 s. Pro výstup z fotoaparátu jsem nastavil kvalitu obrazu na RAW + BASIC, to znamená, že vedle snímku ve formátu NEF získám i snímek ve formátu JPEG s kompresním poměrem 1:16. Ten najde využití k rychlé orientaci mezi jednotlivými snímky, resp. jejich sadami pro jednotlivé měřené pozice. [10]



*Obr. 16 Schéma měření*

Fotoaparát jsem umístil na optickou lavici, kterou jsem upevnil na stativ. Takto upravený stativ jsem umístil do vzdálenosti  $d = 2 \text{ m}$  od displeje tak, aby v poloze  $0 \text{ mm}$  byl fotoaparát v ose displeje (viz Obr. 16). Před displej jsem položil černou látku, aby snížila odraz světla od stolu, a celou místnost jsem zatemnil. Měřil jsem v rozsahu  $400 \text{ mm}$ , tedy  $200 \text{ mm}$  na každou stranu od osy, s krokem  $25 \text{ mm}$ . Celkový počet měřících bodů byl tedy  $17$ . V každém měřícím bodě jsem pořídil  $16$  snímků, z toho  $14$  pro jednotlivé pohledy a  $2$  kalibrační. K měření jsem použil vzor prokládání získaný pomocí metody I (viz kapitola 5.1), který se nejvíce blížil ideálnímu testovacímu obrazci.



*Obr. 17 Ukázka snímků z měření přeslechu, zde pro měřící pozici  $x = -150$   
 (a) pohled č.3, (b) pohled č.4, (c) pohled č.5, (d) pohled č.6*

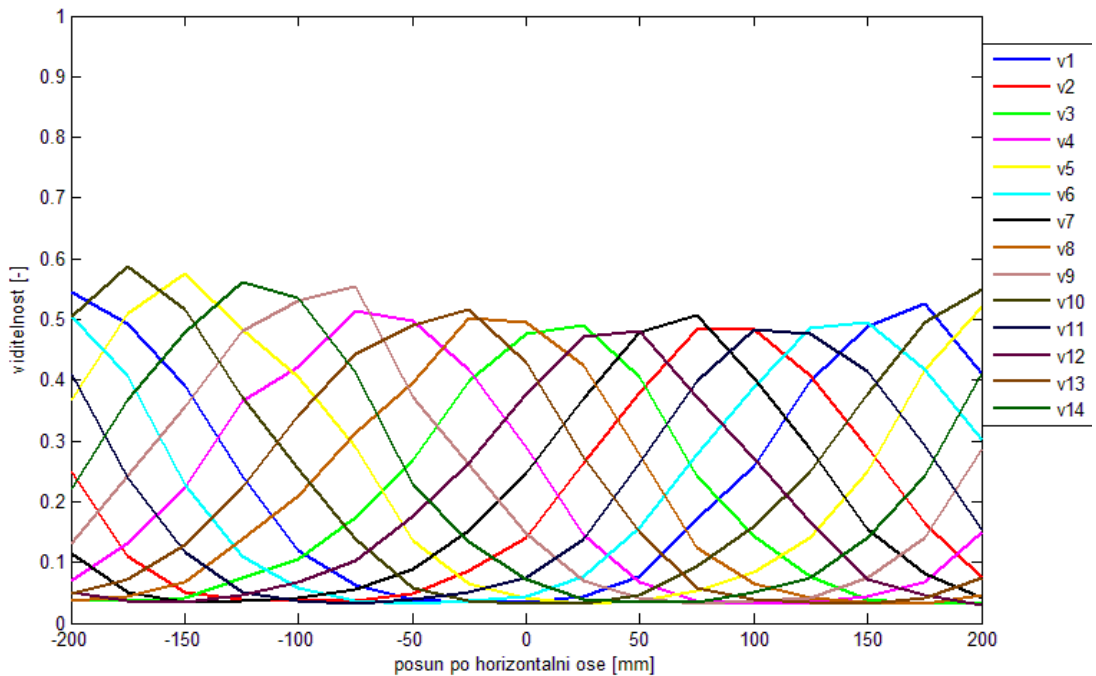
### 6.3 Vyhodnocení výsledků

Pro vyhodnocení jsem použil připravený skript v programu MATLAB, který spočítal jas pořízených snímků (6), kalibroval jej (7) a spočítal *viditelnost* (8) jednotlivých pohledů, tu pak vynesl v grafu viditelnosti (viz Obr. 18).

$$L_{snímku} = \sum_x \sum_y L_{x,y} \quad (6)$$

$$L_{kalibrační} = L_{max} - L_{min} = L_1 - L_2 \quad (7)$$

$$viditelnost = \frac{L_{snímku} - L_{min}}{L_{kalibrační}} \quad (8)$$



Obr. 18 Graf viditelnosti jednotlivých pohledů v závislosti na posuvu podél horizontální osy

Na Obr. 18 můžeme vidět graf viditelnosti jednotlivých pohledů, na kterém je vidět, že jednotlivé pohledy mají průběh podobný Gaussově křivce, což odpovídá předpokladům. Viditelnost jednotlivých pohledů dosahuje hodnot mezi 0,5 – 0,6, jelikož jsme ji kalibrovali s hodnotou jasu při plném rozsvícení displeje, kde se samozřejmě projeví i příspěvky jasu ostatních pohledů. Tento pokles také může být způsoben právě použitím optické lavice, kde jsem se posouval po přímce a ne po kružnici, i když tato chyba v měřicí vzdálenosti by neměla být větší jak 1%. Dále si například můžeme povšimnout, že pohledy

se začínají opakovat po vzdálenosti kratší, než je 400 mm, ale vzhledem k provedení měření považují tuto vzdálenost za vhodně zvolenou.

Pro výpočet přeslechu jsem si hodnoty viditelnosti proložil Gaussovou křivkou, abych získal větší množství hodnot, zejména v oblasti maxim jednotlivých pohledů. Odečetl jsem hodnoty na ose x pro maxima jednotlivých pohledů a pomocí funkcí *feval* a *cfit* vypočítal hodnoty všech pohledů ve vybraných hodnotách na ose x, kde se nacházejí středy daných pohledů. Poté jsem vypočítal přeslech dle vzorce

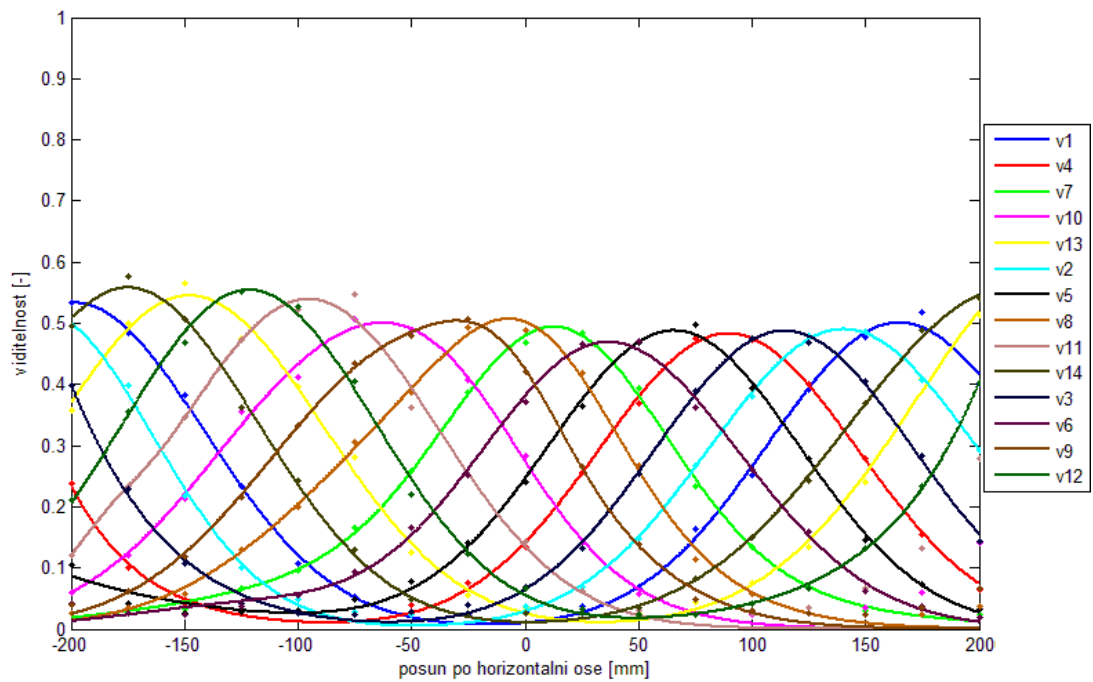
$$XT_{i,j} = \frac{\text{viditelnost}_j}{\text{viditelnost}_{\max,i}}, \quad (9)$$

kde *i* a *j* jsou čísla pohledů, které porovnáváme. Výsledný přeslech jsem vyčíslil v procentech v následující tabulce.

Tabulka přeslechů														
Pohled číslo	10	5	14	9	4	13	8	3	12	7	2	11	6	1
1	87,9	65,9	38,6	19,6	6,4	1,7	1,5	3,3	9,1	23,4	44,7	70,1	93,2	-
2	19,3	8,3	3,6	2,0	3,3	10,9	22,8	39,7	67,2	90,7	-	90,1	66,8	40,5
3	5,2	8,3	12,3	19,6	40,3	71,4	90,7	-	94,6	60,9	36,0	18,8	9,5	5,1
4	22,3	42,4	63,2	83,0	-	86,0	62,4	41,4	21,8	7,6	2,9	0,9	0,2	0,1
5	87,9	-	88,9	68,5	38,9	15,0	6,6	3,2	2,3	4,7	10,9	22,5	42,2	66,4
6	68,9	39,7	16,9	5,9	1,3	1,6	4,2	9,5	22,3	45,0	70,5	91,3	-	89,7
7	10,8	7,5	5,3	4,8	9,1	23,4	41,7	63,4	91,1	-	91,8	68,2	40,9	20,0
8	5,7	13,3	24,7	40,8	68,7	92,8	-	94,9	72,8	35,7	17,2	7,5	2,9	1,0
9	41,8	65,8	88,0	-	91,7	55,9	32,7	18,5	8,1	2,2	0,6	0,1	0,0	0,0
10	-	91,4	64,7	39,5	16,3	4,7	2,3	2,4	5,2	12,5	24,6	42,1	65,5	87,1
11	39,4	19,6	9,0	4,3	2,3	4,3	9,9	20,0	41,0	69,2	92,2	-	89,7	65,4
12	4,2	6,6	8,4	11,3	22,9	47,3	68,9	87,0	-	85,0	64,1	40,2	20,8	9,2
13	10,7	24,3	42,7	65,1	92,1	-	90,7	69,9	41,8	17,6	7,9	3,0	0,9	0,3
14	62,6	91,2	-	92,3	63,7	28,9	14,0	7,2	4,1	4,2	6,9	11,9	21,3	36,6

Tab. 3 Tabulka přeslechů mezi jednotlivými pohledy, hodnoty jsou uvedeny v procentech

V tabulce máme v řádcích průběhy jednotlivých pohledů, zatímco ve sloupcích můžeme číst hodnoty přeslechu v daných středech označených pohledů. Pokud budeme porovnávat tabulku s grafem (Obr. 18), střed pohledu č.10 je na hodnotě  $x_{10} = -175 \text{ mm}$ , poslední hodnota pro střed pohledu č.1 je  $x_1 = 164,8 \text{ mm}$ . Dále bychom mohli vykreslit průběh proložený Gaussovou křivkou a změnit označení pohledů tak, jak je vidíme v prostoru.



*Obr. 19 Graf viditelnosti proložený Gaussovou křivkou a změněným označením pohledů*

Z takto upraveného grafu bychom mohli zpětně sestavit vzor prokládání (viz Příloha 9.2), ve kterém uvidíme různé závislosti, jako například posun jednotlivých pohledů v řádku. Ten se v tomto případě rovná 5. Tedy pokud pohled č.1 bude na subpixelu 1, pak pohled č.2 bude na subpixelu 6 a pohled č.3 na subpixelu 11. Tyto údaje lze využít pro další měření, kde bychom jistě získali lepší výsledky.



## 7 Závěr

Pro měření přeslechu jsem použil vzor prokládání získaný z metody I, která je obecně jednodušší a lze na ni snadno pozorovat základní princip multiview displeje. Metodu II pro zjištění vzoru prokládání jsem nepoužil z důvodu náročného analyzování výsledků. Metoda III je velmi náročná na výpočetní výkon, což mi znemožnilo tuto metodu provést vícekrát s různými parametry, zejména ve fázi vyhodnocení počtu pohledů a sestavování vzoru prokládání. Pokud se však podíváme na výsledné vzory prokládání, můžeme mezi nimi najít některé podobnosti. To samé pak platí o výsledném grafu viditelnosti (Obr. 18), kde jsem si označil pohledy podle toho, jak jsem je budil na displeji, ale jejich pozice v prostoru je jiná. S označení pohledů podle jejich pozice v prostoru jsem vykreslil graf viditelnosti (Obr. 19), kde jsem změřené body proložil Gaussovou křivkou. Jak z obou těchto grafů, tak i z tabulky přeslechů (Tab. 3) můžeme odečíst přeslechy mezi jednotlivými pohledy. Přeslech mezi sousedními pohledy se pohybuje v rozmezí hodnot 87-95%, pro pohledy ob 1 se hodnota přeslechu pohybuje v rozmezí 60-70%. Jelikož jsem použil vzor prokládání s 14 pohledy, ačkoliv displej dle dokumentace podporuje 28 pohledů, výsledný graf pravděpodobně zobrazuje dvojice pohledů, které si jsou velmi podobné. To je příčinou vysokých poměrů přeslechu mezi sousedními pohledy. Další podíl na tomto přeslechu může mít skutečnost, že jednotlivé pohledy se úmyslně překrývají tak, aby při pozorování docházelo k plynulému přechodu mezi těmito pohledy, což by rušilo náš vjem při sledování nějaké scény na displeji. Vzhledem k tomu, že vzdálenost mezi středy jednotlivých pohledů se pohybuje v rozmezí 25 až 30 mm, což je přibližně polovina vzdálenosti mezi lidskýma očima. Z toho tedy vyplývá, že při pohledu na displej uvidíme jedním okem např. pohled č.1 a druhým okem pohled č.3, pokud bychom pracovali s výsledky z tohoto měření. Kdybychom počítali s možností 28 pohledů, vzdálenost mezi maximy by byla poloviční, z čehož vyplývá, že bychom viděli pohledy č.1 a č.5. Mohli bychom také uvažovat, že displej využívá přeslechu k zvýšení rozlišení jednotlivých pohledů, jelikož předpokládá, že scéna na dvou, případně až třech sousedních snímcích se nebude příliš lišit, a tak se obraz může složit z několika takových podobných snímků.

## 8 Seznam použité literatury a zdrojů informací

- [1] Boev, A.; Bregovic, R.; Gotchev, A., "Measurements and modeling of angular visibility in multiview display", *Special issue on 3D display, Journal of Society for Information Display*, Sept. 2010 Vol. 18, No. 09, pp. 686-697
- [2] Urey, H.; Chellappan, K.V.; Erden, E.; Surman, P., "State of the Art in Stereoscopic and Autostereoscopic Displays," *Proceedings of the IEEE*, vol.99, no.4, pp.540,555, April 2011
- [3] Liyuan Xing; Junyong You; Ebrahimi, T.; Perkis, A., "Stereoscopic quality datasets under various test conditions," *Quality of Multimedia Experience (QoMEX), 2013 Fifth International Workshop on*, vol., no., pp.136,141, 3-5 July 2013
- [4] Boev, A.; Gotchev, A.; Egiazarian, K., "Crosstalk Measurement Methodology for Auto-Stereoscopic Screens," *3DTV Conference, 2007*, vol., no., pp.1,4, 7-9 May 2007
- [5] Xiao-Fang Li; Qiong-Hua Wang; Da-Hai Li; Ai-Hong Wang, "Image Processing to Eliminate Crosstalk Between Neighboring View Images in Three-Dimensional Lenticular Display," *Display Technology, Journal of*, vol.7, no.8, pp.443,447, Aug. 2011
- [6] Zhou, M.; Wang, H.; Li, W.; Jiao, S.; Hong, T.; Wang, S.; Sun, X.; Wang, X.; Kim, J.-Y.; Nam, D., "A Unified Method for Crosstalk Reduction in Multiview Displays," *Display Technology, Journal of*, vol.10, no.6, pp.500,507, June 2014
- [7] Lingsheng Kong; Guang Jin; Xing Zhong, "An autostereoscopic projecting system based on parallax barrier and lenticular sheets," *Multimedia Technology (ICMT), 2011 International Conference on*, vol., no., pp.4816,4819, 26-28 July 2011
- [8] BDL4251VS Addendum User Manual. [online]. [cit. 2014-05-22] Dostupné z WWW:  
<http://www.dimenco.eu/downloads/manuals/BDL4251VS%20Addendum%20User%20Manual-English%2020130109.pdf>
- [9] Dimenco Control Tool – User Manual. [online]. [cit. 2014-05-22] Dostupné z WWW:  
[http://www.dimenco.eu/downloads/manuals/DimencoControlTool\\_Manual\\_v3.pdf](http://www.dimenco.eu/downloads/manuals/DimencoControlTool_Manual_v3.pdf)
- [10] Nikon: The Nikon Guide to Digital Photography with the D70 Digital Camera. [online]. [cit. 2014-05-22] Dostupné z WWW: [http://cdn-10.nikon-cdn.com/pdf/manuals/dslr/D70\\_en.pdf](http://cdn-10.nikon-cdn.com/pdf/manuals/dslr/D70_en.pdf)

# 9 Přílohy

## 9.1 Obsah datového nosiče

- Text práce ve formátu PDF
- Skripty v programu MATLAB pro generování snímků a vyhodnocení jednotlivých měření
- Snímky z měření přeslechu ve formátech NEF a JPEG

## 9.2 Vzory prokládání

Pixel 1			Pixel 2			Pixel 3			Pixel 4			Pixel 5	
R	G	B	R	G	B	R	G	B	R	G	B	R	G
2	3	4	5	6	7	8	9	10	11	12	13	14	1
2	3	4	5	6	7	8	9	10	11	12	13	14	1
3	4	5	6	7	8	9	10	11	12	13	14	1	2
3	4	5	6	7	8	9	10	11	12	13	14	1	2
4	5	6	7	8	9	10	11	12	13	14	1	2	3
4	5	6	7	8	9	10	11	12	13	14	1	2	3
5	6	7	8	9	10	11	12	13	14	1	2	3	4
5	6	7	8	9	10	11	12	13	14	1	2	3	4
6	7	8	9	10	11	12	13	14	1	2	3	4	5
6	7	8	9	10	11	12	13	14	1	2	3	4	5
7	8	9	10	11	12	13	14	1	2	3	4	5	6
7	8	9	10	11	12	13	14	1	2	3	4	5	6
8	9	10	11	12	13	14	1	2	3	4	5	6	7
8	9	10	11	12	13	14	1	2	3	4	5	6	7
9	10	11	12	13	14	1	2	3	4	5	6	7	8
9	10	11	12	13	14	1	2	3	4	5	6	7	8
10	11	12	13	14	1	2	3	4	5	6	7	8	9
10	11	12	13	14	1	2	3	4	5	6	7	8	9
11	12	13	14	1	2	3	4	5	6	7	8	9	10
11	12	13	14	1	2	3	4	5	6	7	8	9	10
12	13	14	1	2	3	4	5	6	7	8	9	10	11
12	13	14	1	2	3	4	5	6	7	8	9	10	11
13	14	1	2	3	4	5	6	7	8	9	10	11	12
13	14	1	2	3	4	5	6	7	8	9	10	11	12
14	1	2	3	4	5	6	7	8	9	10	11	12	13
14	1	2	3	4	5	6	7	8	9	10	11	12	13
1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	2	3	4	5	6	7	8	9	10	11	12	13	14

Tab. 4 Vzor prokládání vytvořený pomocí metody 1

Pixel 1			Pixel 2			Pixel 3			Pixel 4			Pixel 5	
R	G	B	R	G	B	R	G	B	R	G	B	R	G
4	7	10	13	2	5	8	11	14	3	6	9	12	1
4	7	10	13	2	5	8	11	14	3	6	9	12	1
7	10	13	2	5	8	11	14	3	6	9	12	1	4
7	10	13	2	5	8	11	14	3	6	9	12	1	4
10	13	2	5	8	11	14	3	6	9	12	1	4	7
10	13	2	5	8	11	14	3	6	9	12	1	4	7
13	2	5	8	11	14	3	6	9	12	1	4	7	10
13	2	5	8	11	14	3	6	9	12	1	4	7	10
2	5	8	11	14	3	6	9	12	1	4	7	10	13
2	5	8	11	14	3	6	9	12	1	4	7	10	13
5	8	11	14	3	6	9	12	1	4	7	10	13	2
5	8	11	14	3	6	9	12	1	4	7	10	13	2
8	11	14	3	6	9	12	1	4	7	10	13	2	5
8	11	14	3	6	9	12	1	4	7	10	13	2	5
11	14	3	6	9	12	1	4	7	10	13	2	5	8
11	14	3	6	9	12	1	4	7	10	13	2	5	8
14	3	6	9	12	1	4	7	10	13	2	5	8	11
14	3	6	9	12	1	4	7	10	13	2	5	8	11
3	6	9	12	1	4	7	10	13	2	5	8	11	14
3	6	9	12	1	4	7	10	13	2	5	8	11	14
6	9	12	1	4	7	10	13	2	5	8	11	14	3
6	9	12	1	4	7	10	13	2	5	8	11	14	3
9	12	1	4	7	10	13	2	5	8	11	14	3	6
9	12	1	4	7	10	13	2	5	8	11	14	3	6
12	1	4	7	10	13	2	5	8	11	14	3	6	9
12	1	4	7	10	13	2	5	8	11	14	3	6	9
1	4	7	10	13	2	5	8	11	14	3	6	9	12
1	4	7	10	13	2	5	8	11	14	3	6	9	12

Tab. 5 Vzor prokládání získaný zpětným sestavením dle výsledků měření přeslechu