

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

AUTOMATICKÁ SEGMENTACE PERIODICKÉHO POHYBU SRDEČNÍHO SVALSTVA V ULTRAZVUKOVÉM ZÁZNAMU

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

MILAN MUNZAR

BRNO 2015



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

AUTOMATICKÁ SEGMENTACE PERIODICKÉHO POHYBU SRDEČNÍHO SVALSTVA V ULTRAZVUKOVÉM ZÁZNAMU

AUTOMATIC SEGMENTATION OF CARDIAC TISSUE MOVEMENT FROM ULTRASOUND

RECORD

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

MILAN MUNZAR

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. ŠTĚPÁN MRÁČEK

BRNO 2015

Abstrakt

Tato práce se zabývá návrhem a implementací metody časové segmentace srdečních stahů v echokardiografickém záznamu. Navržený postup je postaven na algoritmech Lucas-Kanade a rychlé Fourierovy transformace. Navržená metoda segmentace je implementována v jazyce C++ za použití knihoven OpenCV a FFTW. Analýza přesnosti segmentace touto metodou ukazuje její náchylnost na kvalitu echokardiografických snímků a výskyt anomálií. Nicméně tyto nedostatky je možno odstranit použitím několika diskutovaných technik. Tato metoda je vyvíjena jako součást projektu pro analýzu echokardiografických záznamů pro Fakultní nemocnici u sv. Anny v Brně.

Abstract

This thesis describes design and implementation of method, which determines beginning of heart beats in echocardiographic record. Design of this method is built around pyramidal Lucas-Kanade algorithm and fast Fourier transform. This method is implemented in C++ language with OpenCV and FFTW libraries. Analysis of the implementation has shown, that this method is sensitive to anomalies in echocardiographic record. This method is developed as a part of the project for an analysis of echocardiographic records for st. Anna hospital at Brno.

Klíčová slova

fyziologie srdce, echokardiografický záznam, časová segmentace, optický tok, metoda Lucas-Kanade, Fourierova transformace

Keywords

heart physiology, echocardiographic record, time segmentation, optical flow, Lucas-Kanade method, Fourier transform

Citace

Milan Munzar: Automatická segmentace periodického pohybu srdečního svalstva v ultrazvukovém záznamu, diplomová práce, Brno, FIT VUT v Brně, 2015

Automatická segmentace periodického pohybu srdečního svalstva v ultrazvukovém záznamu

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením pana Ing. Štěpána Mráčka.

.....
Milan Munzar
2. června 2015

Poděkování

Děkuji vedoucímu práce panu Ing. Štěpánu Mráčkovi za jeho návrhy, připomínky a diskuzi. Také bych chtěl poděkovat panu Doc. Janu Černockému za jeho konzultaci ke zpracování signálu.

© Milan Munzar, 2015.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1 Úvod	2
2 Fyziologie lidského srdce	3
2.1 Čerpací funkce srdce	3
2.2 Tvorba a vedení srdečního vzruchu	4
2.3 Kontrakce srdečního svalu	6
2.4 Řízení srdeční činnosti	6
2.5 Echokardiografické vyšetření srdce	7
2.6 Popis EKG křivky	8
3 Segmentace srdečních stahů v echokardiografickém záznamu	10
3.1 Výpočet optického toku	10
3.2 Řešení optického toku metodou Lucas-Kanade	11
3.3 Iterativní verze metody Lucas-Kanade	13
3.4 Výpočet Lucas-Kanade nad pyramidovou strukturou	14
3.5 Využití metody Lucas-Kanade k segmentaci srdečních stahů	17
3.6 Určení periody signálu pomocí diskretní Fourierovy transformace	20
3.7 Algoritmus segmentace echokardiografického záznamu	23
4 Implementace segmentace echokardiografického záznamu.	27
4.1 Implementace metody Lucas-Kanade	27
4.2 Testování parametrů optického toku	29
4.3 Implementace diskretní Fourierovy transformace	30
4.4 Implementace rekonstrukce EKG křivky	31
5 Měření přesnosti algoritmu segmentace	33
5.1 Měření přesnosti jednotlivých záznamů	33
5.2 Poznatky získané z měření přesnosti	37
6 Závěr	41
A Test parametrů optického toku	44
B Obsah DVD	46

Kapitola 1

Úvod

Tato diplomová práce popisuje návrh a implementaci metody, která automaticky rozdělí echokardiografický záznam na jednotlivé srdeční stahy. Implementovaná metoda je součástí projektu, který se zabývá vývojem softwaru pro analýzu srdeční činnosti z echokardiografických záznamu. Tento software bude používán ve Fakultní nemocnici u sv. Anny v Brně.

Druhá kapitola této práce se zabývá stavbou a funkcí lidského srdce (kapitola 2). Je zde popsána struktura srdečního svalu, jehož levou komoru pozorujeme na echokardiografických záznamech a jednotlivé srdeční fáze systoly a diastoly, podle kterých je záznam segmentován. Stručně je zde popsána tvorba a vedení srdečního vzruchu, který způsobuje svalovou kontrakci a tím vypuzuje krev ze srdce do krevního oběhu. Dále jsou zde popsány různé typy echokardiografických vyšetření, jejich průběh a účel. Poslední odstavce pojednávají o EKG vyšetření. Díky dostupnosti EKG křivek u jednotlivých echokardiografických záznamů je možno validovat implementovanou metodu pro segmentaci echokardiografických záznamů.

Třetí kapitola se zabývá návrhem metody segmentace srdečních stahů (kapitola 3). Je zde popsána metoda optického toku Lucas-Kanade, která dovoluje sledovat pohyb objektů mezi jednotlivými snímky videa. Pro zpřesnění metody Lucas-Kanade se používá její iterativní forma, která se aplikuje na pyramidovou strukturu jednotlivých echokardiografických snímků. Dále tato kapitola popisuje jak získat charakteristiku pohybu ze sledování srdeční stěny metodou Lucas-Kanade. Po získání této charakteristiky lze echokardiografický záznam segmentovat. Výpočet hranic segmentů je proveden analýzou frekvenčního spektra této charakteristiky pomocí rychlé Fourierovi transformace.

Čtvrtá kapitola popisuje implementaci navržené metody pro segmentaci echokardiografických snímků (kapitola 4). Implementace využívá prostředků jazyka C++, knihovny OpenCV a knihovny FFTW. Prvně zde popisujeme použití metody Lucas-Kanade v prostředí knihovny OpenCV. V následující sekci se zabýváme implementací diskrétní Fourierovy transformace pomocí prostředků knihovny FFTW. V poslední části je popsána implementace rekonstrukce EKG křivky z dostupných echokardiografických záznamů.

V páté kapitole je měřena úspěšnost implementované metody pro časovou segmentaci echokardiografického záznamu (kapitola 5). Je zde popsána použitá metodologie založená na mapování hranic vypočtených segmentů na odpovídající pozice na EKG křivce. Tímto je možno určit v jaké srdeční fázi dochází k segmentaci a ověřit tak, že jednotlivé segmenty jsou správně zarovnány vzhledem k srdeční fázi.

Závěr poté rekapituluje dosažené výsledky (kapitola 6). Zejména hodnotí jak úspěšně probíhala segmentace na echokardiografických snímcích. Také jsou zde navržena možná vylepšení, která jsou nezbytná pro nasazení tohoto algoritmu do nemocničního prostředí.

Kapitola 2

Fyziologie lidského srdce

Srdce je dutý svalový orgán, jehož hlavní funkcí je zajištění cirkulace krve v oběhové soustavě. Oběhová soustava je tvořena cévami a umožňuje transport živin a kyslíku tkáním. Současně odvádí odpadní látky k orgánům umožňujícím jejich vyloučení. K tomu, aby krev proudila v cévách je nutná práce srdce, která se spotřebovává na tření proudící kapaliny [16]. Tato práce se dá přirovnat ke dvěma sériově zapojeným pumpám, které vhánějí krev do dvou sériově zapojených oběhů [20]. První pumpou je levá část srdce napojená aortou na velký oběh. Velký oběh zásobuje orgány okysličenou krví a následně krev vrací zpět žilami do pravé části srdce. Z pravé části srdce se krev pumpuje plicnicí do malého oběhu, kde se při průchodu plicemi okysličí. Poté se vrací zpět do levé části srdce (obr. 2.1). Srdce je tedy rozděleno na levou a pravou polovinu, každá naráz pumpující krev do svého oběhu.

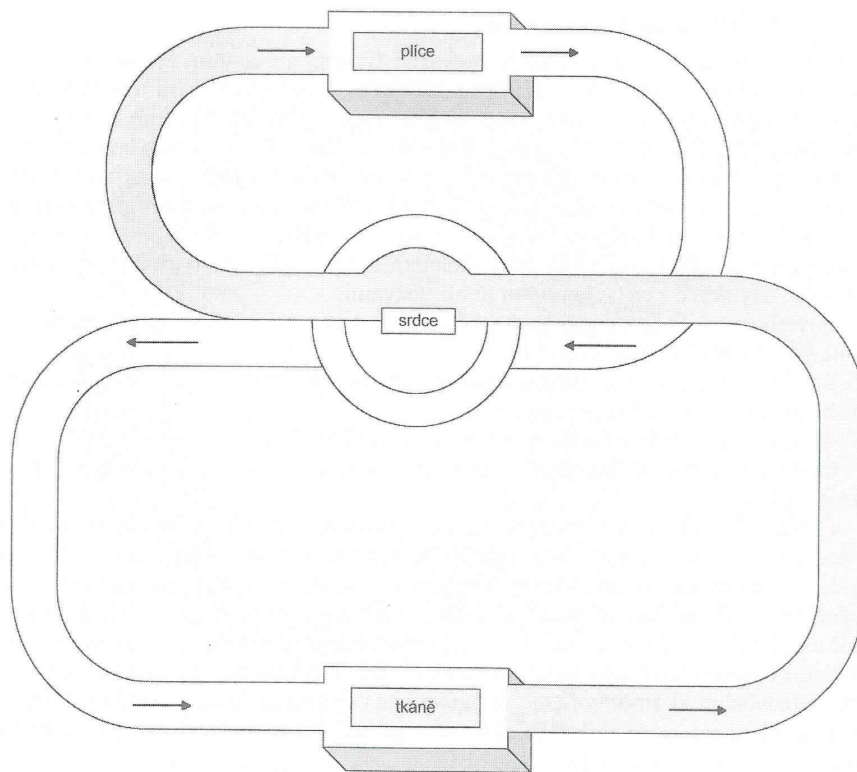
Srdeční sval je tvořen čtyřmi dutinami: pravé komory a předsíně a levé komory a předsíně. Levá komora má výrazně silnější srdeční stěnu než pravá, protože pumpuje krev aortou proti mnohem většímu odporu ve velkém oběhu [15]. Mezi předsíněmi a komorami a mezi komorami a velkými tepnami (aortou a plicnicí) se nachází chlopně, které řídí směr průtoku krve. Srdeční chlopeň se otevírá nebo uzavírá podle toho, jaký je poměr tlaku na obou stranách chlopně.

Vnější část srdečního svalu se nazývá perikard, který chrání srdce před třením a nárazy. Prostřední část srdeční stěny je tvořena myokardem. Myokard je ze všech tří částí srdeční stěny nejsilnější a umožňuje tvorbu srdečních stahů. Vnitřní část stěny poté tvoří endokard, jenž vystýlá srdeční dutinu a je v kontaktu s protékající krví. Na povrchu srdce se nachází srdeční základna. To je místo kam vstupují nebo odkud vystupují cévy. Celé srdce váží asi 0.5% celkové hmotnosti člověka [20].

2.1 Čerpací funkce srdce

Čerpací funkce srdce vzniká rytmickým střídáním srdečního stahu (systola) a jeho relaxací (diastola). Během diastoly se srdce plní krví, během systoly se krev pumpuje do velkých tepen. Srdeční komory jsou plněny ze srdečních předsíní kam se krev dostává z velkých žil. Fáze diastoly je zřetelně delší než fáze systoly [20].

Střídání systoly a diastoly lze podrobněji popsat fázemi srdeční revoluce [16]. První fáze systoly je fáze napínací, která začíná kontrakcí komor. Všechny chlopně (cípaté i poloměsíčné) jsou v této fázi uzavřeny a tlak v komorách tak velmi rychle stoupá. Po překonání tlaku ve velkých tepnách dochází k otevření poloměsíčných chlopní mezi nimi. Nastává fáze vypuzovací. Během této fáze dosahuje tlak v komorách a velkých tepnách svého maxima.



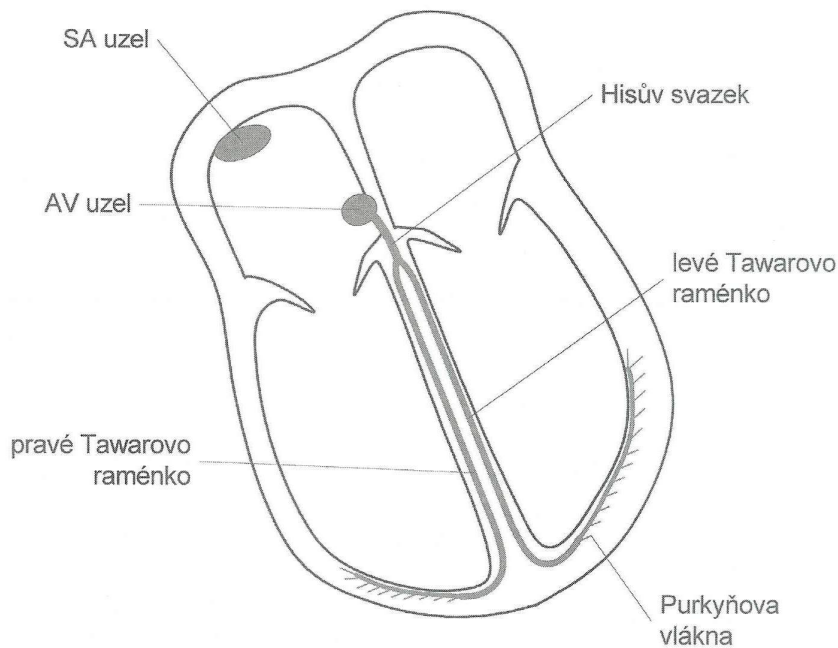
Obrázek 2.1: Schéma oběhové soustavy člověka. Převzato z [20].

Na počátku dochází k vypuzení většiny tepového výdeje. Poté začíná tlak v komorách klesat, až nakonec klesne pod hodnotu tlaku velkých tepen. Poloměsíčné chlopně se uzavírají a v této fázi také dochází k plnění předsíní. Nyní začíná diastola svoji fází relaxace. Tlak v komorách prudce klesá, zatímco tlak v předsíních stoupá a cípaté chlopně mezi nimi se otevírají. Začíná plnicí fáze. Krev proudí z předsíní do komor podobně jako u fáze vypuzovací nejprve velice rychle a poté se plnění zpomaluje. Nakonec se kontrahují předsíně čímž se doplní komory.

Minutový srdeční výdej udává množství krve přečerpané z jedné komory do příslušné tepny. Je vypočítán vynásobením tepové frekvence se systolickým objemem a představuje v klidu přibližně 70 tepů za minutu \times 80ml, tedy asi 5.6l za minutu [19]. Při námaze se tato hodnota zvyšuje až na 35l (u vrcholových sportovců až 40l) za minutu. Nejvýznamějším faktorem pro zvyšování minutového srdečního výdeje je zvyšování tepové frekvence. Zvyšování frekvence srdce však neznamená pouze více tepů za jednotku času, ale také se mění poměr mezi dobou trvání systoly a diastoly. Zatímco systola se zkracuje jen velmi málo, diastola se zkracuje podstatně [20].

2.2 Tvorba a vedení srdečního vzruchu

Buňky srdečního svalu, podobně jako u svalu kosterního a hladkého, dokáží reagovat na podráždění jejich membrány vyvoláním elektrické odpovědi zvané akční potenciál. Akční potenciál představuje informaci, kterou tyto buňky nesou a mohou ji předávat dále. Podstatou jeho vzniku je otevření iontových kanálů závislých na napětí [16]. Změna napětí na mem-



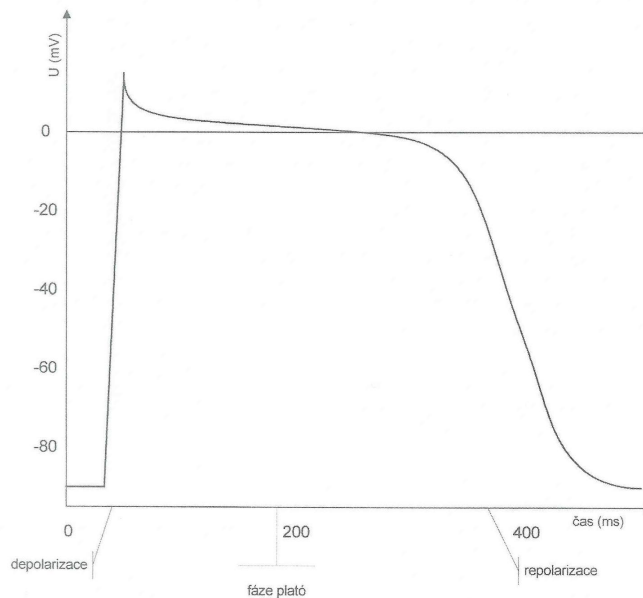
Obrázek 2.2: Převodní systém srdeční. Převzato z [20].

bráně způsobuje otevření Na^+ iontových kanálů, sodík může tak proudit do nitra buňky, což mění polaritu membrány. Membrána se stává pozitivnější oproti klidovému stavu. Tento děj se nazývá depolarizace. Současně, ale mnohem pomaleji se otevírají K^+ iontové kanály. Draslík po otevření proudí ven z buňky. To pozitivitu snižuje opět do negativních hodnot, obnovuje se původní polarita membrány (repolarizace).

Akční potenciál se u jednotlivých buněk myokardu liší svým průběhem, především podle toho jakou funkci daná buňka zastává (zda-li se jedná o buňku pracovního myokardu nebo buňku převodního systému srdečního), ale také podle její lokalizace [20]. Převodního systému srdečního samovolně vytváří a vede elektrický vzruch v náležité posloupnosti po celém srdci (obr. 2.2). Je tvořen sinoatriálním uzlem, který je za normálních okolností zdrojem vzruchu. Jeho charakteristickou vlastností je, že má nejrychlejší depolarizaci a tedy než ostatní buňky dosáhnou své spouštěcí úrovně, již obdrží vzruch od sinoatriálního uzlu. Vzruch se zněj dále šíří na obě předsíně k atrioventrikulárnímu uzlu. Odtud dospěje přes Hisův svazek a obě Tawarova raménka až k Purkyňovým vláknům, která přenesou vzruch na myokard komor. Vzruch, který vytváří sinoatriální uzel udává rytmus celému srdci a označuje se jako sinusový rytmus.

Buňky pracovního myokardu nejsou schopny spontánně rytmické vzruchy vytvářet, ale odpovídají na vzruch kontrakcí svalů. Akční potenciál je u těchto buněk charakteristiký především dlouhou dobou jejich trvání, která je dána fází plateau (obr. 2.3). Během této fáze vstupuje do buněk především kalcium Ca^{++} , a tak udržují kladný náboj uvnitř buňky. Díky dlouhé depolarizaci těchto buněk každý nový akční potenciál vyvolá nový svalový stah. Tím se zabraňuje vzniku tetanického stahu, protože se již nezvyšuje síla kontrakce původního stahu [16].

Protože si srdce tyto rytmické podněty vytváří samo hovoříme o tzv. automacii. Jakmile kdekoliv v srdci vznikne dostatečné podráždění, vyvolá vzruchovou aktivitu, která se rozšíří soubunním myokardu po celém srdci. Srdce tedy vždy odpovídá jako celek, neboť rozhraní



Obrázek 2.3: Akční potenciál pracovního myokardu. Převzato z [20]

mezi jednotlivými buňkami tvoří interkalární disky, jenž mají velice nízký odpor [20].

2.3 Kontrakce srdečního svalu

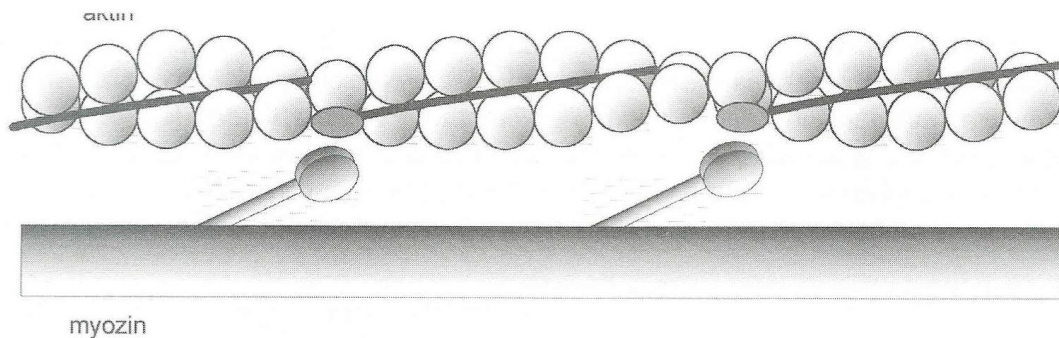
Srdeční stah vzniká vzájemným posunem aktinových a myosinových vláken proti sobě. Aktinová vlákna jsou tvořena dvěma řetězci kulovitých aktinových monomerů, které jsou mezi sebou vzájemně obtočeny (obr. 2.4). V pravidelných odstupech jsou mezi oběma aktinovými vlákny umístěny molekuly troponinu spojené vláknitým troponinem. Molekula myosinu má tvar tenké tyčinky s kulovitým rozšířením. Svazek těchto molekul tvoří myozinové vlákno, na kterém se periodicky objevují zmíněná kulovitá rozšíření vyčnívající proti vláknům aktinu [20].

Pokud není v buňkách myokardu dostatek iontů Ca^{++} , zabraňují tropomyozinová vlákna svým umístěním vzniku vazby mezi myosinem a aktinem. Během depolarizace stoupá koncentrace Ca^{++} uvnitř buňky. Ionty vápníku reagují s troponinem, troponin změnil konformaci a odkryjí se vazebná místa pro myosinové hlavy na aktinových vláknech. Ty se do svých vazebných míst zasunou a za rozštěpení ATP nastává ohnutí myosinové hlavy, které způsobí posun aktinových vláken proti myosinovým [20].

Svalová kontrakce i relaxace srdečního svalu vyžaduje energii. Zdrojem energie je štěpení nukleotidu ATP. Myokard na rozdíl od kosterního svalstva nemůže pracovat na kyslíkový dluh. Využívá však ve větší míře jiné zdroje energie (kys. mléčnou, neesterifikované mastné kyseliny) [16].

2.4 Řízení srdeční činnosti

Srdce musí přizpůsobovat svou činnost měnícím se potřebám organismu změnami tepové frekvence, tepového objemu a krevního tlaku. Toto řízení je jednak zevní (hormonální a nervové), tak i vnitřní.



Obrázek 2.4: Struktura aktinových a myosinových vláken. Převzato z [20].

Srdeční frekvence je řízena téměř výlučně nervově. Sinoatriální uzel, který je za normálních okolností za srdeční frekvenci odpovědný je pod neustálým vlivem vegetativního nervového systému - sympatiku a parasympatiku. Působením sympatiku se srdeční frekvence zvyšuje a naopak působení parasympatiku srdeční frekvenci tlumí. Vegetativní nervový systém má také vliv na zvyšování tepového objemu. Principem zvyšování je stimulace Ca^{++} kanálů a tím i zvětšením průniku Ca^{++} iontů během fáze plató do srdečních buněk. Parasympatikus má účinky opačné a pracuje tak, že potlačuje činnosti sympatiku.

Hormonální řízení se uskutečňuje pomocí katecholaminů (adrenalin a noradrenalin) a acetylcholinu. Prostřednictvím těchto látek působí vegetativní nervový systém na srdce. Stimulace některým z katecholaminů má analogický efekt k působení sympatiku. Naopak působením acetylcholinu vyvolává potlačující efekt obdobný při působení parasympatiku.

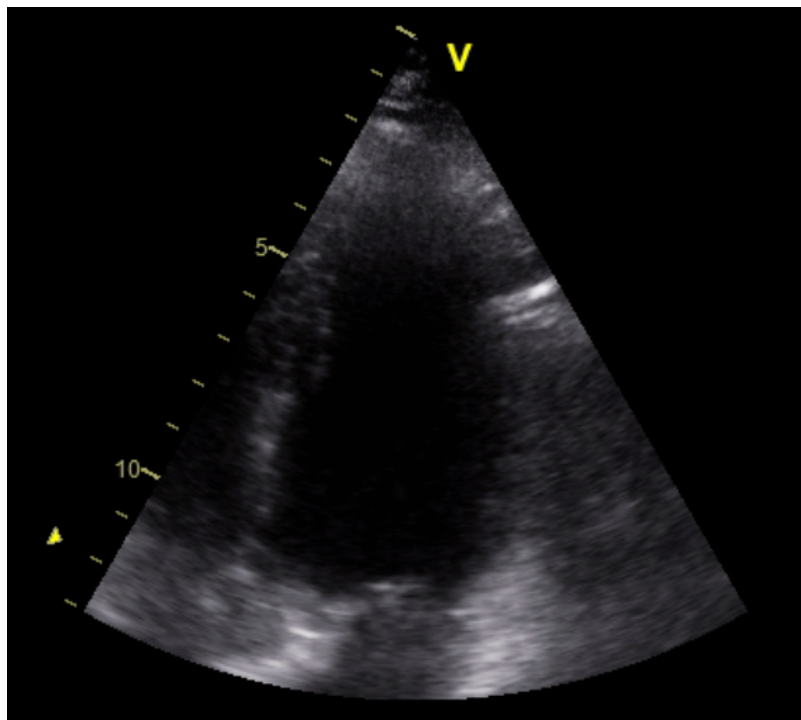
Mezi místní mechanismy srdeční činnosti se řadí Starlingův zákon, který říká, že síla srdeční kontrakce se zvětšuje se zvyšujícím se plicním objemem. To je způsobeno tím, že když se vlákna více protáhnou, počet možných interakcí aktinu a myosinu se zvětší a stah se zesílí [16]. Tímto zákonem je zaručena rovnováha mezi přečerpaným objemem u obou komor srdce. Stoupne-li např. tepový objem pravé komory, zvýší se množství krve přitékající do levé komory. Zvětšení náplně v levé komoře způsobí zvětšení jejího tepového objemu a tím se objem obou komor vyrovná.

2.5 Echokardiografické vyšetření srdce

Echokardiografie je neinvazivní vyšetření srdce pomocí ultrazvukových vln. Výstupem vyšetření je záznam, který zachycuje pozici a pohyb srdečních struktur. Tento záznam je většinou zobrazován v kuželové struktuře (obr. 2.5), která přibližně kopíruje tvar srdce otočeného hrotem nahoru. Čitelnost záznamu zpřesňují míry, které jsou po stranách záznamu.

Echokardiogram, přiložený na hrudník, vysílá zvukové vlnění o frekvenci 2.5 - 10 Mhz. Zvukové vlnění proniká do vyšetřované oblasti, kde se odráží a část zvukových vln se vrací zpět do echokardiogramu. Počítač poté vypočítá vzdálenost struktury od níž se signál odrazil z času mezi odesláním a přijmutím ultrazvuku (ultrazvuk se šíří konstantní rychlostí) a podle intenzity odraženého signálu zobrazí danou strukturu ve stupních šedi na obrazovce. Místa, od kterých se ultrazvuk odrazí kompletně, se zobrazí bíle. Místa, kterými ultrazvuk projde, se zobrazí černě.

Dopplerovská echokardiografie navíc umožňuje zobrazit proudění krve v srdci a velkých cévách pomocí Dopplerova efektu. Proudění krve se navenek projevuje mimo jiné vznikem vlnění v oblasti ultrazvukových frekvencí, přičemž konkrétní hodnota frekvence je závislá



Obrázek 2.5: Levá komora zobrazená na echokardiografickém záznamu.

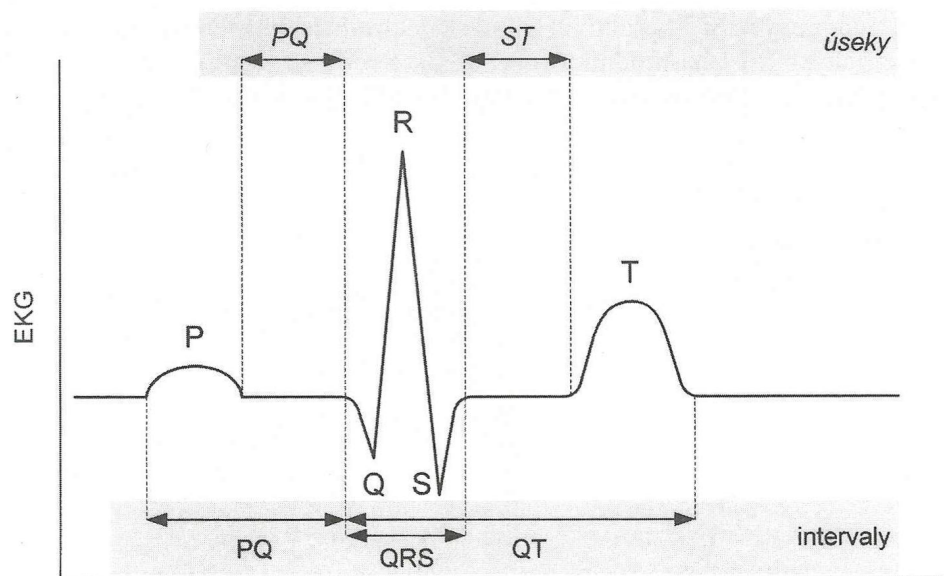
na rychlosti proudu [20].

Echokardiografické vyšetření trvá několik minut a pro pacienta je bezbolestné a bez známých vedlejších efektů. Pacient v průběhu vyšetření leží s obnaženým hrudníkem na zádech nebo na boku. Výsledky jsou známy ihned po vyšetření. Lékař může ze záznamu zjišťovat např. velikost srdce a jednotlivých srdečních oddílů, charakter srdečního pohybu, nedomikavost chlopní, či zda srdce vypuzuje dostatečné množství krve. U dětí může echokardiografie odhalit vrozené srdeční vady.

2.6 Popis EKG křivky

Elektrická aktivita srdečních buněk se projevuje změnami elektrického napětí na povrchu těla. Protože jednotlivé srdeční buňky nemají ve stejném okamžiku stejný akční potenciál a nepracují zcela synchronně (sekce 2.2), tečou po membránách a v okolí srdce elektrické proudy [20]. Jelikož jsou lidské tkáně vodivé, můžeme na různých místech těla naměřit různá napětí, která jsou projevem šíření akčního potenciálu myokardem. Průběh těchto napětí je znázorněn EKG křivkou jako rozdíly potenciálů (mV). Z této křivky následně můžeme zjistit polohu srdce, fyzický stav, tepovou frekvenci, vzruchové rytmicity a původ a šíření elektrického vzruchu [19]. Avšak narozdíl od echokardiogramu nelze určit kontrakční a čerpací práce srdce.

Průběh elektrokardiografického vyšetření je většinou neinvazivní. Elektrickou aktivitu srdce snímáme z povrchu těla svorkami, které jsou tvořeny elektrodami. Původní způsob měření zavedl Willem Einthoven. Toto měření se provádí pomocí tří bipolárních svodů umístěných do trojúhelníku na rukách a levé noze [19]. Dnes je měření zdokonaleno přidáním dalších devíti svodů. Jedná se o unipolární svody umístěné jednak na končetinách a také na hrudi. U unipolárních svodů platí pravidlo - pokud vlna depolarizace směřuje k elektrodě,



Obrázek 2.6: Elektrokardiografická křívka. Převzato z [20].

registruje se na ni kladná výchylka, směřuje-li vlna depolarizace od elektrody registrujeme zápornou výchylku. U repolarizace toto pravidlo platí obráceně. U bipolárních svorek platí analogická pravidla viz [20].

Na naměřené EKG křívce (obr. 2.6) rozeznáváme vlny P,T a kmity Q,R a S. Vzdálenosti mezi vlnami a kmity nazýváme úseky. Úseky spolu s vlnami tvoří intervaly. Vlna P je výrazem depolarizace předsíní. Vzruch se šíří ze sinoatriálního uzlu a vlna depolarizace se rozšíří na svalovinu předsíní. Amplituda vlny P je malá, neboť tenká stěna předsíní obsahuje poměrně málo svaloviny. Když dospěje vlna depolarizace do atrioventrikulárního uzlu (obr. 2.2) dojde k její zbrždění. Toto zbrždění je dáno strukturou atrioventrikulárního uzlu a jeho účelem je zajistit, že systola předsíní bude dokončena před systolou komor. Na EKG se v této fázi zapíše úsek PQ.

Následně přechází vzruch přes Hissov svazek a Tawarova raménka na oblast mezikomorového septu [20]. Zde vyvolá depolarizaci, která se šíří ve směru od levé komory k pravé komoře. Tuto událost znázorňuje kmit Q. Vzruch se pak šíří dále a vyvolává depolarizaci v oblasti srdečního hrotu, což je znázorněno kmitem R. Následně se provádí depolarizace svaloviny komor a to ve směru od endokardu k epikardu (kmit S). Tento sled událostí se na EKG křívce nazývá QRS komplex.

Po rozšíření depolarizace po celé svalovině komor nastává fáze plateau (sekce 2.2). Elektrická aktivita srdce je v této fázi nulová. EKG tento stav znázorňuje úsekem ST. Na fázi plateau navazuje repolarizace komor, která narozdíl od depolarizace postupuje od epikardu k endokardu. Tuto fázi představuje vlna T. Protože je však repolarizace děj elektricky opačný, mívá vlna T na většině svorkách kladnou amplitudu.

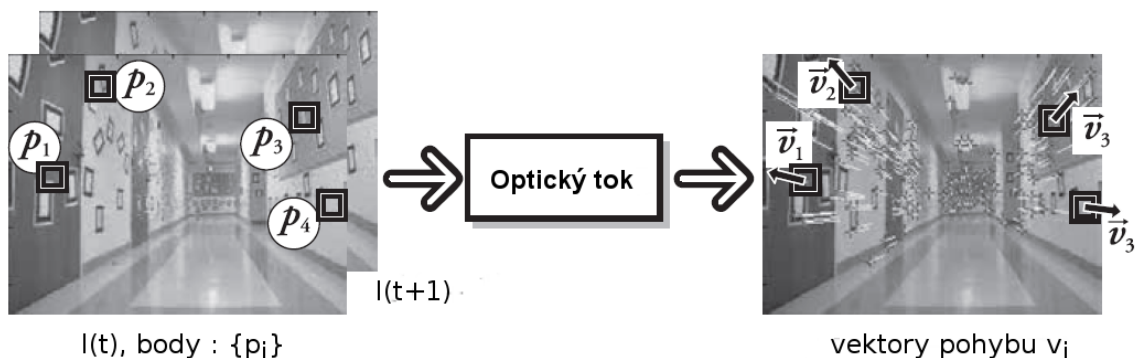
Kapitola 3

Segmentace srdečních stahů v echokardiografickém záznamu

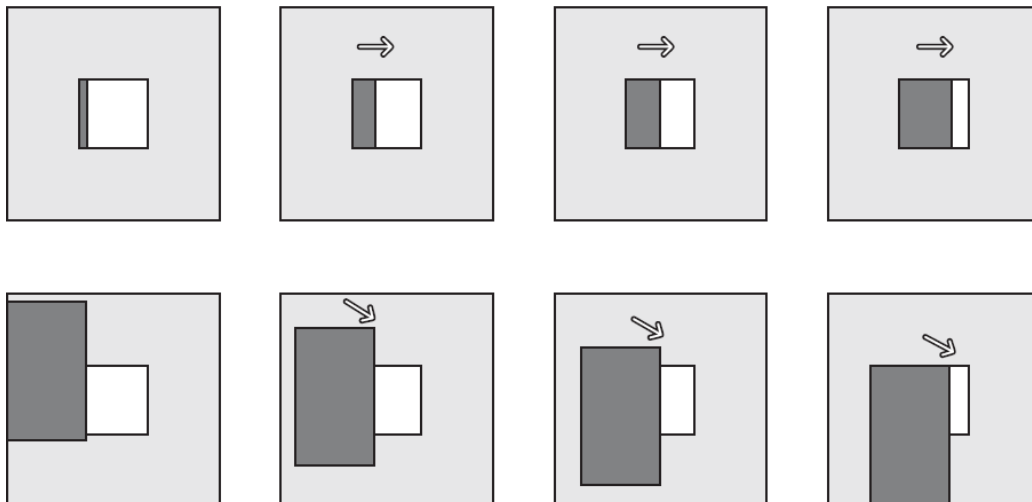
Informaci o pohybu srdeční stěny v echokardiografickém záznamu můžeme získat metodami optického toku. Tyto metody zachycují změnu umístění jednotlivých pixelů vzhledem k následujícímu snímku ve video sekvenci (obr. 3.1). Aplikujeme-li metodu optického toku na oblast srdeční stěny v echokardiografickém záznamu, získáme informaci o pohybu srdce. Obdržený signál poté můžeme rozložit na jednotlivé frekvenční složky pomocí Fourierovy transformace. Z těchto frekvenčních složek poté vybereme tu s nejnižší frekvencí a spočítáme periodu srdečních stahů. Poté segmentujeme daný echokardiografický záznam vzhledem k této periodě a obdržíme tak časový údaj o výskytu systolických stahů v záznamu.

3.1 Výpočet optického toku

Výpočet optického toku hledá k jednotlivým pixelům obrázku změnu jejich pozice vzhledem k následující obrázku ve videu. Pro následující výpočet předpokládám 2D obrázky v odstupech šedi a hledanou změnu pozice v x-ovém směru označím u a změnu y-ovým směru v . Prvním předpokladem výpočtu optického toku je, že intenzita pixelů nějakého objektu ve videu se nemění [3]. Pokud funkci intenzity rozšíříme o časovou složku t a dt jako změnu snímku ve videu, můžeme první předpoklad zapsat takto:



Obrázek 3.1: Pohyb zachycený metodou optického toku Lucas-Kanade. Převzato z [3].



Obrázek 3.2: *Aperture problem*. V horním řádku sledujeme pohyb objektu skrze malé okénko. V dolním řádku vidíme skutečný pohyb objektu. Převzato z [3].

$$I(x, y, t) = I(x + u, y + v, t + dt) \quad (3.1)$$

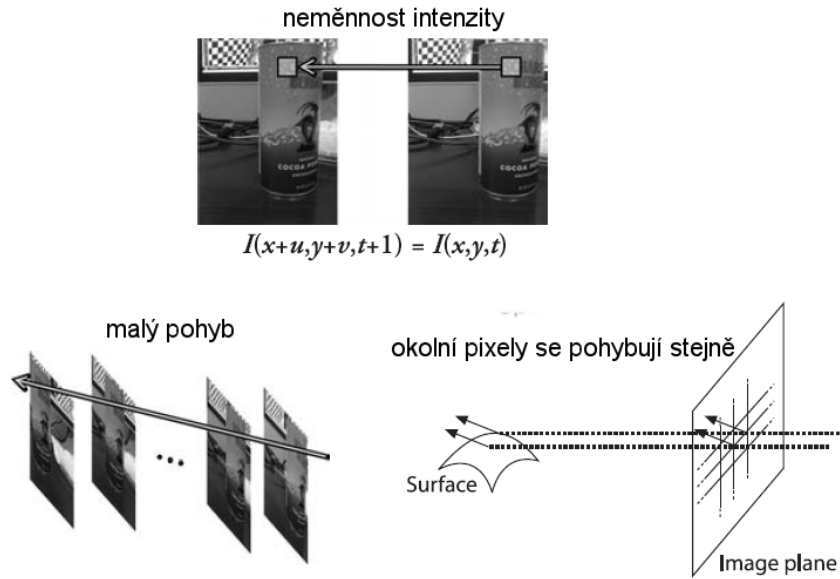
Druhý předpoklad hovoří o tom, že pohyb objektu mezi dvěma následujícími obrázky ve videu je pouze malý [3]. Díky tomuto předpokladu si můžeme vyjádřit gradienty změny intenzit pixelů ve videu a odhadnout tak směr pohybu. Aplikujeme-li tedy tyto dva předpoklady (aplikace Taylorova rozvoje na pravou stranu rovnice a následné upravy) obdržíme rovnici optického toku (rovn. 3.2) [12]. Složky I_x , I_y a I_t jsou gradienty intenzit v příslušných směrech. Neznámé rovnice u, v vyjadřují velikost pohybu v daném směru.

$$I_x u + I_y v + I_t = 0 \quad (3.2)$$

Vidíme, že rovnici optického toku (rovn. 3.2) nelze spočítat, protože obsahuje dvě neznámé. Tato vlastnost se u optického toku označuje jako *aperture problem* a hovoří o tom, že pokud sledujeme pohyb nějakého objektu skrze malé okénko (v našem případě jeden pixel), tak nemůžeme přesně určit pohyb celého objektu (obr. 3.2) [3]. K tomu abychom mohli tuto rovnici úspěšně řešit zavádíme dodatečná omezení jako například v [12] a [14].

3.2 Řešení optického toku metodou Lucas-Kanade

Metoda Lucas-Kanade zavádí k rovnici optického toku 3.2 třetí předpoklad (obr. 3.3) a umožňuje ji tak úspěšně řešit metodou nejmenších čtverců [14]. Třetí předpoklad hovoří o tom, že blízké okolí sledovaného pixelu se sním pohybuje stejně. To nám umožňuje sestavit pro každý uvažovaný bod $p_1 \dots p_n$ rovnici optického toku a mít tak dodatečná omezení pro výpočet u, v . Tato úvaha se dá zapsat následovně:



Obrázek 3.3: Předpoklady úspěšného výpočtu metody Lucas-Kanade. Převzato z [3].

$$\underbrace{\begin{bmatrix} I_x(p_1) & I_y(p_1) \\ I_x(p_2) & I_y(p_2) \\ \vdots & \vdots \\ I_x(p_n) & I_y(p_n) \end{bmatrix}}_{\substack{A \\ n \times 2}} \underbrace{\begin{bmatrix} u \\ v \end{bmatrix}}_{\substack{d \\ 2 \times 1}} = - \underbrace{\begin{bmatrix} I_t(p_1) \\ I_t(p_2) \\ \vdots \\ I_t(p_n) \end{bmatrix}}_{\substack{b \\ n \times 1}} \quad (3.3)$$

Stále ovšem neumíme z rovnice 3.3 vyjádřit hledané u, v . Problémem je, že máme více rovnic než neznámých a tedy daná soustava rovnic nemusí mít řešení. Avšak alespoň přibližné řešení můžeme nalézt metodou nejmenších čtverců, kde hledáme $\min \|Ad - b\|^2$ [3]. Aplikujeme-li metodu nejmenších čtverců na rovnici 3.3 získáváme následující vztah:

$$(A^T A)d = A^T b \quad (3.4)$$

Rozepíšeme-li rovnici 3.4 detailněji obdržíme:

$$\underbrace{\begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix}}_{\substack{A^T A \\ 2 \times 2}} \begin{bmatrix} u \\ v \end{bmatrix} = - \underbrace{\begin{bmatrix} \sum I_x I_t \\ \sum I_y I_t \end{bmatrix}}_{\substack{A^T b \\ 2 \times 1}} \quad (3.5)$$

Složky u, v potom obdržíme vyřešením následující rovnice:

$$\begin{bmatrix} u \\ v \end{bmatrix} = (A^T A)^{-1} A^T b \quad (3.6)$$

Rovnice 3.6 má řešení pouze tehdy, je-li matice $A^T A$ invertibilní (musí být čtvercová a zároveň regulární). Pěkná vlastnost této podmínky je, že je splněna v těch částech obrázku, kde textura směřuje alespoň do dvou směrů (snímek obsahuje dva a více vlastních

vektorů) [3]. Pokud sledujeme ve výřezu videa například rohový objekt můžeme přesně odhadnout jeho směr pohybu. To nám konečně řeší *aperture problem*. Stejný princip je použit u Harrisova detektoru hran [9].

Bohužel samotná metoda Lucas-Kanade nepracuje příliš přesně tam, kde je veliký pohyb mezi snímky videa (neplatí druhý předpoklad). Můžeme sice zvětšovat uvažované okolí, nicméně to nám může narušit předpoklad, že se jednotlivé pixely okolí budou pohybovat stejně. Proto se používá její iterativní forma aplikovaná na pyramidovou strukturu jednotlivých snímků videa (sekce 3.4).

3.3 Iterativní verze metody Lucas-Kanade

Iterativní verze metody Lucas-Kanade nadále zlepšuje přesnost optického toku. Je podobná metodě tečen, která se používá pro hledání kořenů soustav nelineárních rovnic. V jedné iteraci metody Lucas-Kanade je metodou nejmenších čtverců nalezen pohyb bodu mezi snímky videa. Iterativní forma několikrát tento výpočet zopakuje a zpřesní aproximaci řešení tím, že použije výsledek z předchozí aproximace jako startovní bod následujícího výpočtu.

Rovnice intenzity, která zohledňuje počáteční odhad v_x^k, v_y^k v kroku k bude vypadat následovně:

$$I(x, y, t) = I(x + v_x^k + u, y + v_y^k + v, t + dt) \quad (3.7)$$

Řešení této rovnice je popsáno v sekci 3.2. Pěkná vlastnost při iteračním výpočtu optického toku je, že gradienty intenzit jednotlivých bodů okolí (matice $A^T A$) stačí spočítat pouze jednou. Zůstávají neměnné v průběhu výpočtu [2]:

$$I_x(x, y) = \frac{I(x + 1, y) - I(x - 1, y)}{2} \quad (3.8)$$

$$I_y(x, y) = \frac{I(x, y + 1) - I(x, y - 1)}{2} \quad (3.9)$$

Jediné co je nutné počítat znovu je matice $A^T b$, která vyjadřuje odlišnost okolí bodů na snímcích při translaci okolí bodu o v_x^k, v_y^k . Každý nový odhad $\vec{v}^k = [v_x^k \ v_y^k]^T$ je vypočítán následovně:

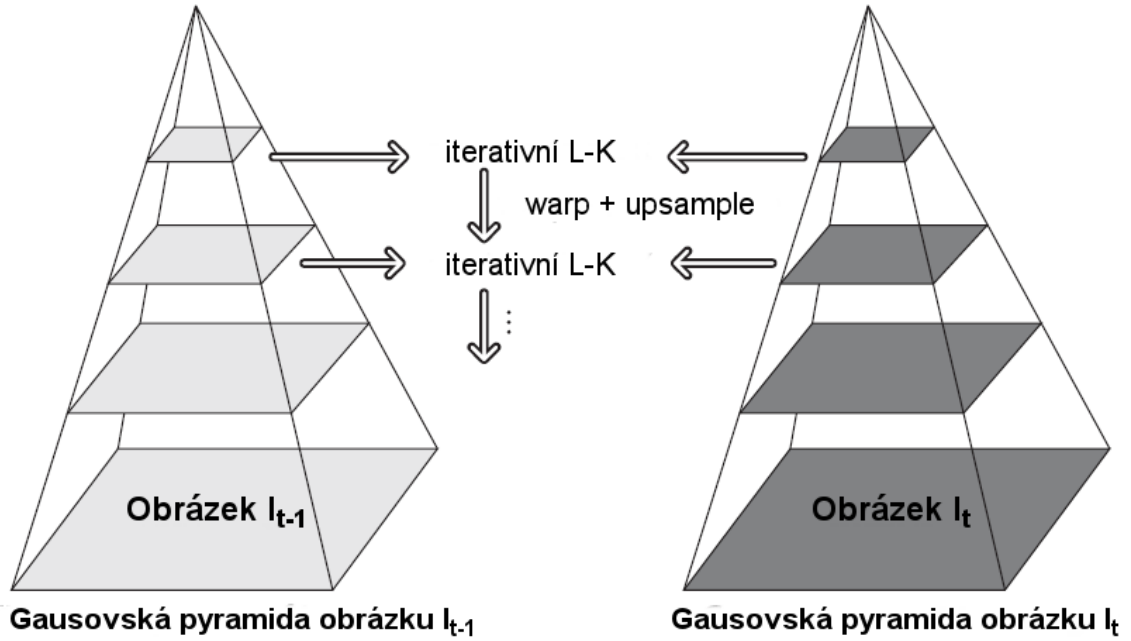
$$\vec{v}^k = \vec{v}^{k-1} + \begin{bmatrix} u \\ v \end{bmatrix} \quad (3.10)$$

Iterování výpočtu zastavíme pokud nově vypočtený vektor $d^k = [u \ v]^T$ je menší než zvolený práh (například menší než 0.1 pixelu) nebo pokud dosáhneme zvoleného maximálního počtu iterací. V první iteraci je počáteční odhad nastaven na nulu:

$$\vec{v}^0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (3.11)$$

Pokud jsme k dosažení řešení konvergence potřebovali K iterací, pak řešení optického toku pro hledaný bod je vektor \vec{v} :

$$\vec{v} = \vec{v}^K = \sum_{k=1}^K d^k \quad (3.12)$$



Obrázek 3.4: Aplikace iterativní formy algoritmu Lucas-Kanade na pyramidovou strukturu snímků. Výsledek každé iterace metody Lucas-Kanade je použit jako startovní bod pro následující iteraci. Převzato z [3].

3.4 Výpočet Lucas-Kanade nad pyramidovou strukturou

Iterativní verze metody Lucas-Kanade aplikovaná na pyramidovou strukturu snímků dovolu je zachytit pohyb bodů ve videu i pokud je větší než velikost zvoleného okolí prohledávání. Pyramidová struktura obsahuje patra snímků, kde každé následující patro obsahuje zmenšený snímek. Touto strukturou poté postupně procházíme od nejvyššího patra a na každé patro aplikujeme iterativní metodu Lucas-Kanade (sekce 3.3). Výsledek jednoho výpočtu optického toku je pak použit jako startovní bod pro výpočet v následujícím patře (obr. 3.4).

Konstrukci takové Gausovské pyramidy (existuje také Laplasova pyramida, která se konstruuje v opačném směru viz [1]), se provádí rekurzivně od nejnižšího patra tvořené originálním snímkem jeho postupným zmenšováním o $1/4$. Tedy každé následující patro pyramidy P^{i+1} je vytvořeno konvolucí Gausovského jádra s předchozím patrem P^i a následným odstraněním lichých sloupců i řádků [2]:

$$\begin{aligned}
 P^{i+1} = & \\
 & \frac{1}{4}P^i(2x, 2y) + \frac{1}{8}(P^i(2x - 1, 2y) + P^i(2x + 1, 2y) + P^i(2x, 2y - 1) + P^i(2x, 2y + 1)) + \\
 & \frac{1}{16}(P^i(2x - 1, 2y - 1) + P^i(2x + 1, 2y + 1) + P^i(2x - 1, 2y + 1) + P^i(2x + 1, 2y - 1))
 \end{aligned}
 \tag{3.13}$$

Počet zvolených pater pyramidy by měl odpovídat očekávanému maximálnímu pohybu objektů ve videu. V naší aplikaci mají echokardiografické snímky rozlišení 636×434 . Tedy pyramida o čtyřech patrech bude obsahovat snímky o šířce 636×434 , 318×217 , 159×119 , 80×60 a 40×30 .

Jelikož je každý snímek pyramidu různě veliký, tak každý bod $b = [b_x \ b_y]$ nějakého snímku bude na různé pozici v každé vrstvě pyramidu. Uvažme, že pyramida obsahuje m pater $L = 0$ až $L = m$. Podle definice pyramidové struktury (rovn. 3.13) jsou koordinaty bodu b v jednotlivých vrstvách pyramidu spočteny následovně:

$$b^L = \frac{b}{2^L} \quad (3.14)$$

Metoda Lucas-Kanade poté postupuje od nejvyššího patra pyramidu L_0 , kde je snímek s nejmenším rozlišením, až k původnímu snímku ve spodním patře. Pro každý bod původního snímku vypočítá hodnotu optického toku v aktuálním patře a tu poté předá jako startovní odhad nižšímu patru. Jedná se o shodný princip k iterační metodě Lucas-Kanade.

Mějme startovní odhad optického toku pro patro L označen $g^L = [g_x^L \ g_y^L]^T$, který obržíme postupnou propagací výsledků z patra L_m do patra $L + 1$. Pro výpočet optického toku v patře L je nutno najít vektor pohybu $d^L = [u^L \ v^L]^T$, který minimalizuje rozdíl mezi okolními hledaného bodu na obrázcích I, J . Rozšíření o tento odhad z vyššího patra lze opět popsat rovnicí neměnnosti intenzity stejným způsobem jako u rovnice 3.7 nebo lze použít alternativního zápisu:

$$\epsilon^L(d^L) = \sum_{x=b_x^L-w_x}^{b_x^L+w_x} \sum_{y=b_y^L-w_y}^{b_y^L+w_y} (I^L(x, y) - J^L(x + g_x^L + d_x^L, y + g_y^L + d_y^L))^2 \quad (3.15)$$

Všimněme si, že odhad optického toku g^L v rovnici 3.15 nám stejně jako u iterační metody Lucas-Kanade (sekce 3.3) posune výřez druhého obrázku J . Díky tomu je vektor d^L malý a je možné aplikovat metodu Lucas-Kanade na výpočet optického toku bodu b mezi obrázky I, J . Výpočet odhadu pro následující patro $L - 1$ se provede následovně:

$$g^{L-1} = 2(g^L + d^L) \quad (3.16)$$

Vektor optického toku nižšího patra d^{L-1} je poté spočítán shodným postupem. Opět použijeme metodu Lucas-Kanade pro nalezení minima funkce $\epsilon^{L-1}(d^{L-1})$ (rovn. 3.15) a výsledek propagujeme nižšímu patru. Tato procedura se opakuje až po původní obrázek v patře L_0 . Algoritmus je inicializován nastavením odhadu pro patro L_m na nulu:

$$g^{L_m} = [0 \ 0]^T \quad (3.17)$$

Výsledný optický tok je k dispozici po poslední aplikaci algoritmu Lucas-Kanade na spodní patro pyramidu:

$$d = g^0 + d^0 \quad (3.18)$$

Výhodou tohoto výpočtu nad pyramidou je, že vektor d^L je malý i když celkový pohyb d objektů ve videu je velký. Pokud každý krok iterace algoritmu Lucas-Kanade nalézá pohyb o maximální velikosti d_{max} , pak nad pyramidovou strukturou bude nalézat pohyb o velikosti $d_{max_final} = (2^{L_m+1} - 1)d_{max}$. Například pyramida hloubky $L_m = 3$ zvětší velikost d_{max} patnáctkrát. To umožňuje nalézat velký pohyb při zanechání malého sousedního okolí a zachytit tak i velké pohyby mezi snímky videa. Celkové shrnutí výpočtu optického toku iterativní metodou Lucas-Kanade nad pyramidovou strukturou je znázorněno algoritmem 1.

Algoritmus 1: Iterativní Lucas-Kanade nad pyramidovou strukturou.

Cíl: Mějme bod na obrázku I . Najdi jeho odpovídající pozici v na obrázku J .

Postav pyramidovou reprezentaci I a J : $I^L_{L=0,\dots,L_m}$ a $J^L_{L=0,\dots,L_m}$. (rovn. 3.13)

Inicializuj startovní odhad g : $g^{L_m} = [g_x^{L_m} \ g_y^{L_m}] = [0 \ 0]^T$. (rovn. 3.17)

for $L = L_M$ **to** 0 **do**

Pozice bodu u v obrázku I^L : $u^L = [p_x \ p_y]^T = u/2^L$. (rovn. 3.14)

Derivuj I^L s ohledem na x : $I_x(x, y) = \frac{I^L(x+1, y) - I^L(x-1, y)}{2}$. (rovn. 3.8)

Derivuj I^L s ohledem na y : $I_y(x, y) = \frac{I^L(x, y+1) - I^L(x, y-1)}{2}$. (rovn. 3.9)

Matice gradientů G : $G = \begin{matrix} \sum_{p_x-w_x}^{p_x+w_x} \sum_{p_y-w_y}^{p_y+w_y} & \begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix} \end{matrix}$ (rovn. 3.5)

Inicializace iterativního L-K: $\vec{v}^0 = [0 \ 0]^T$. (rovn. 3.11)

for $k = 1$ **to** K **do**

Rozdíl obrázku: $\delta I_k(x, y) = I^L(x, y) - J^L(x + g_x^L + v_x^{k-1}, y + g_y^L + v_y^{k-1})$

Vektor rozdílů obrázků: $\vec{b}_k = \sum_{p_x-w_x}^{p_x+w_x} \sum_{p_y-w_y}^{p_y+w_y} \begin{bmatrix} \delta I_k(x, y) I_x(x, y) \\ \delta I_k(x, y) I_y(x, y) \end{bmatrix}$ (rovn. 3.5)

L-K optický tok: $\vec{\eta}_k = G^{-1} \vec{b}_k$ (rovn. 3.6)

Odhad další iterace: $\vec{v}^k = \vec{v}^{k-1} + \vec{\eta}_k$ (rovn. 3.10)

end

Optický tok na úrovni L : $d^L = \vec{v}^k$

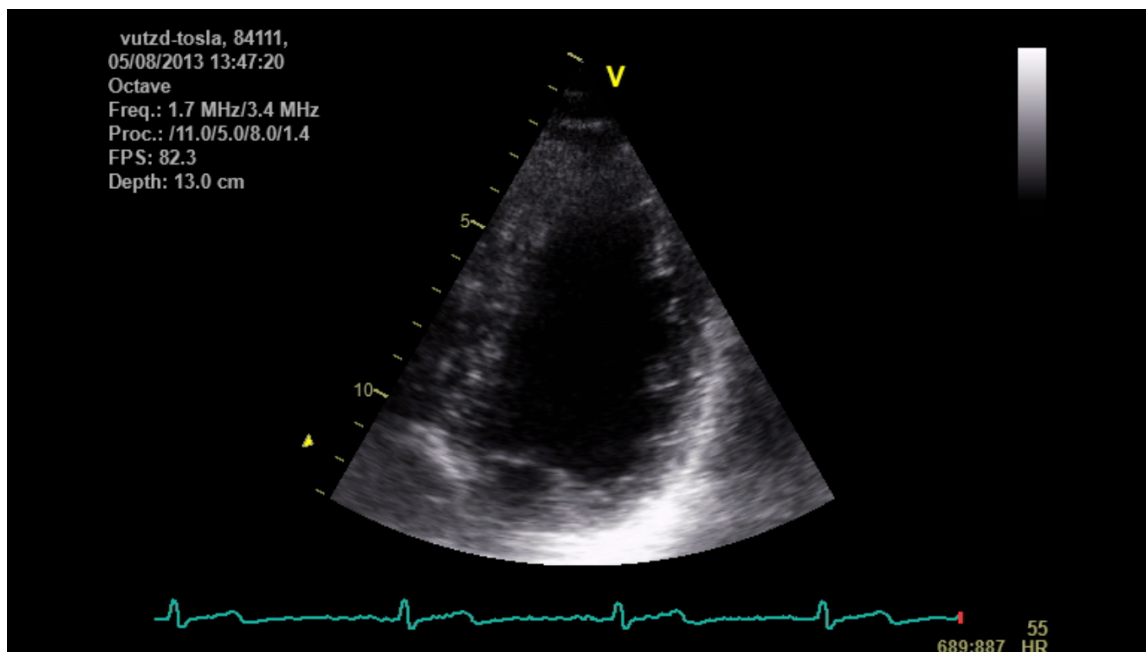
Odhad pro další úroveň $L - 1$: $g^{L-1} = [g_x^{L-1} \ g_y^{L-1}]^T = 2(g^L + d^L)$ (rovn. 3.16)

end

Finální optický tok: $d = g^0 + d^0$ (rovn. 3.18)

Umístění bodu na obrázku J : $v = u + d$

Řešení: Hledaný bod je na pozici v na obrázku J .



Obrázek 3.5: Obrazovka echokardiografického záznamu.

3.5 Využití metody Lucas-Kanade k segmentaci srdečních stahů

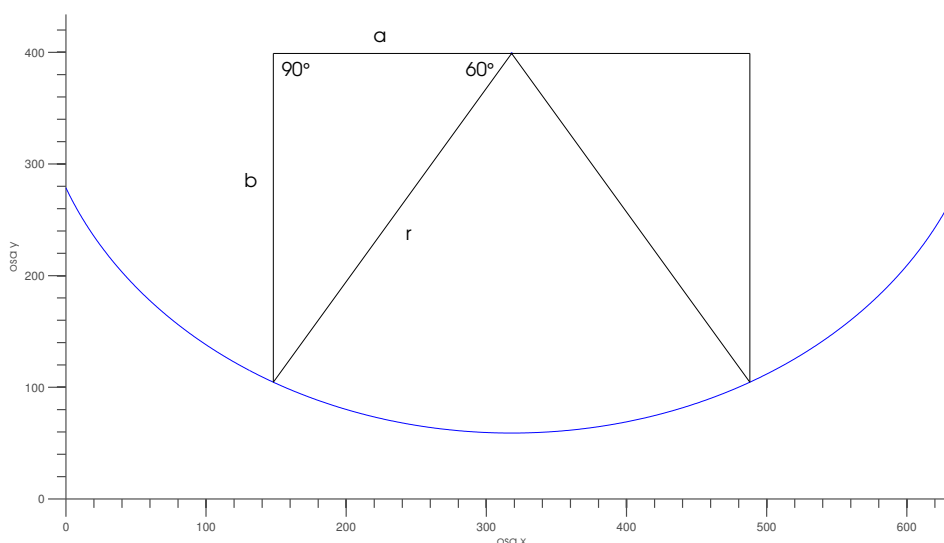
Proto, aby metoda Lucas-Kanade přesně zachytila pohyb srdeční stěny, je nutné jí nejprve vymezit sledovanou oblast echokardiografického záznamu. V poskytnutých záznamech se vyskytuje několik artefaktů, které mají nepříznivý vliv na vyhodnocení charakteru pohybu srdeční stěny touto metodou. Jedná se především o posouvající se EKG křivku v dolní části videa, počítadlo snímků a další údaje navíc viz obrázek 3.5. Metodou Lucas-Kanade tedy budeme sledovat pouze body uvnitř kruhového výřezu echokardiografického záznamu. Z charakteru pohybu těchto bodů spočteme signál, pomocí něhož budeme následně segmentovat daný záznam na jednotlivé srdeční stahy (sekce 3.6).

Začneme s testem, zda-li nějaký bod leží uvnitř nebo vně kruhové výseče. Pro bod ležící uvnitř kruhové výseče musí platit tři následující podmínky:

- Jeho vzdálenost od středu kruhové výseče musí být menší než je její poloměr.
- Musí ležet nalevo od pravého ramene kruhové výseče.
- Musí ležet napravo od levého ramene kruhové výseče.

První podmínka omezení vzdálenosti od středu je triviální. Stačí spočítat délku úsečky dané středem kruhové výseče a uvažovaným bodem pomocí Pythagorovy věty a následně ověřit, zda vypočtená délka nepřesahuje velikost poloměru kruhové výseče.

Pro ověření zbylých dvou podmínek si ramena kruhové výseče vyjádříme ve vektorovém tvaru. Námi dostupné echokardiografické záznamy mají kruhovou výseč o středovém úhlu 60 stupňů a poloměru 340 pixelů. Pokud si promítneme do této výseče pravouhlý trojúhelník jako je tomu na obrázku 3.6, tak můžeme vypočítat jeho odvěsny a , b podle vztahů:



Obrázek 3.6: Výpočet vektoru ramen kruhové výseče.

$$a = \cos(60^\circ) \cdot 340, \quad b = \sin(60^\circ) \cdot 340 \quad (3.19)$$

Z vypočtených odvěsen a, b si vyjádříme vektor \vec{L} odpovídající levému rameni a vektor \vec{P} odpovídající pravému rameni kruhové výseče s následujícími souřadnicemi:

$$\vec{L} = [-a, b], \quad \vec{P} = [a, b] \quad (3.20)$$

Dále k vektorům \vec{L}, \vec{P} definujeme jejich normálové vektory \vec{L}_n a \vec{P}_n směřující dovnitř kruhové výseče:

$$\vec{L}_n = [b, -a], \quad \vec{P}_n = [-b, -a] \quad (3.21)$$

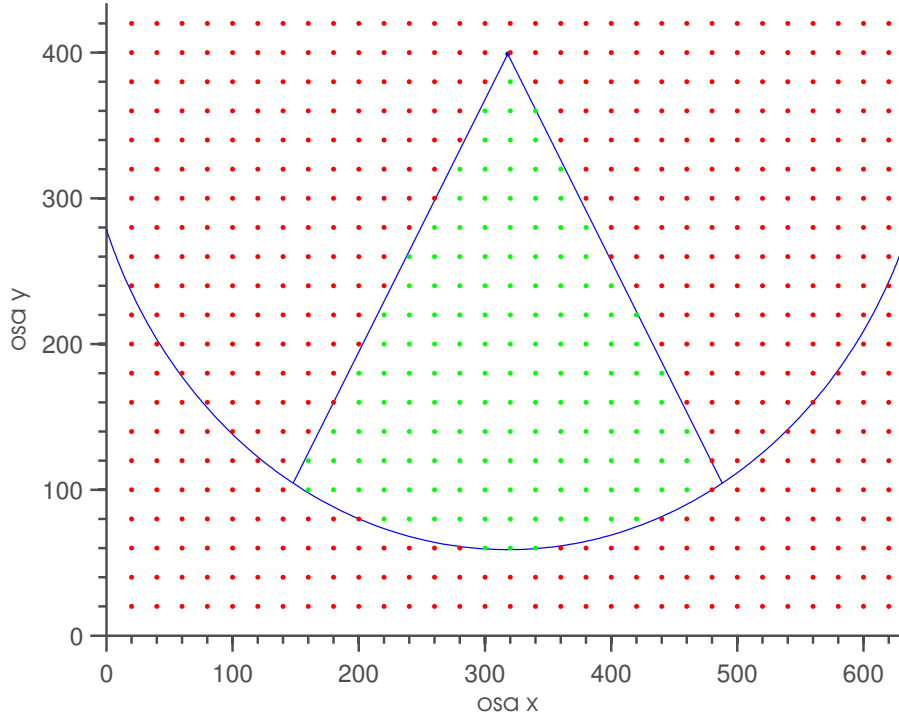
Poslední komponentou, kterou potřebujeme k určení, zda-li daný bod leží uvnitř kruhové výseče, je vektor \vec{V} vyjadřující polohu sledovaného bodu od středu výseče. Tento vektor má následující souřadnice:

$$\vec{V} = [bod_x - stred_x, bod_y - stred_y] \quad (3.22)$$

Provedeme-li nyní skalární součin vektoru \vec{V} s vektorem \vec{L}_n můžeme ze znaménka výsledku určit, na jaké straně levého ramena kruhové výseče leží sledovaný bod. V případě, že skalární součin výjde kladný, pak se sledovaný bod nachází napravo od levého ramene. V případě záporného výsledku leží sledovaný bod nalevo. Tato vlastnost plyne z následující rovnice pro libovolné dva vektory \vec{u}, \vec{v} :

$$\vec{u} \cdot \vec{v} = |\vec{u}| \cdot |\vec{v}| \cdot \cos\alpha \quad (3.23)$$

Všimněme si, že výsledek pravé strany rovnice bude díky kosinu záporný vždy pokud úhel α , který svírají oba vektory, spadá do intervalu od 90 do 270 stupňů. Aplikujeme-li stejné pravidlo zároveň na pravé rameno kruhové výseče (rovn. 3.24), obdržíme test, který určí zda sledovaný bod leží uvnitř kruhové výseče (obr. 3.7).



Obrázek 3.7: Test na umístění bodu uvnitř kruhové výseče.

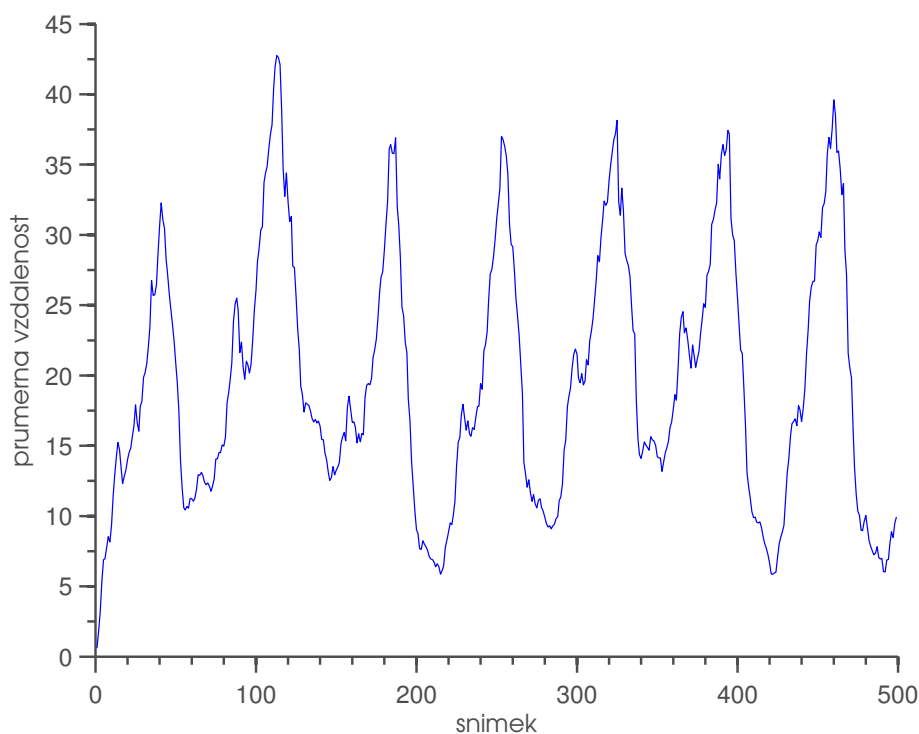
$$(\vec{V} \cdot \vec{L}_n \geq 0) \wedge (\vec{V} \cdot \vec{P}_n \geq 0) \iff \text{leží uvnitř kruhové výseče} \quad (3.24)$$

Omezení sledování pouze těch bodů, které leží uvnitř kruhové výseče, nám nestačí k tomu, abychom odstranily vliv pro nás nepotřebných komponent echokardiografického záznamu. To je způsobeno pyramidovým výpočtem metody Lucas-Kanade (sekce 3.4), kde se ve vyšších patrech uvažuje širší okolí a tak může tato metoda do svého výpočtu zahrnout i okolní rušivé komponenty. Proto jednotlivé snímky echokardiografického záznamu ořízneme obdelníkovým oknem, které bude obsahovat pouze kruhovou výseč. Tento výřez můžeme popsat následovně pomocí souřadnic levého horního bodu a pravého dolního bodu uvážíme-li značení S pro střed kruhu a r pro jeho poloměr:

$$vyrez = [(S_x - a, S_y), (S_x + a, S_y + r)] \quad (3.25)$$

Posledním krokem k aplikaci metody Lucas-Kanade je určení bodů, jejichž pohyb bude touto metodou sledován. V námi zvoleném algoritmu tento krok provedeme tak, že echokardiografický záznam pokryjeme čtvercovou mřížkou bodů o stejných vzdálenostech a nad každým bodem z této mřížky provedeme test, zda leží uvnitř kruhové výseče viz obr. 3.7. Body, jenž jsou vně kruhové výseče dále neuvažujeme a algoritmus Lucas-Kanade aplikujeme pouze na body ležící uvnitř kruhové mřížky.

Vlastní aplikaci metody Lucas-Kanade provedeme tak, že oblasti prvního echokardiografického snímku určené body B postupně hledáme na všech následujících snímcích ve videu. Nejedná se tedy o běžnou aplikaci optického toku sledování pohybu bodu ze snímku na snímek. Tímto způsobem jsme schopni vyhodnotit vzdálenost sledovaných bodů od jejich původního umístění na prvním snímku echokardiografického záznamu. Po každé takové



Obrázek 3.8: Signál získaný metodou Lukas-Kanade.

aplikaci metody Lucas-Kanade spočítáme vzdálenosti všech N bodů od příslušných původních pozic a jako jednu položku signálu vezmeme průměr těchto vzdáleností viz rovn 3.26. Příklad takové aplikace metody Lucas-Kanade je na obrázku 3.8.

$$signal = \frac{\sum_{k=0}^{k=N} |B[k] - B_{lk}[k]|}{N} \quad (3.26)$$

Následné zpracování signálu provedeme pomocí diskrétní Fourierovy transformace za účelem zjistit periodu signálu (sekce 3.6). Echokardiografický záznam posléze segmentuje od oblasti maxima signálu po kroku rovném velikosti periody (sekce 3.7). Jednotlivé intervaly záznamu budou odpovídat srdečním stahům.

3.6 Určení periody signálu pomocí diskrétní Fourierovy transformace

Aplikujeme-li diskrétní Fourierovu transformaci na signál obdržíme jeho frekvenční spektrum. Ze získaného spektra si pomocí spektrální hustoty výkonu určíme dominantní frekvenční složku. Perioda této složky bude odpovídat periodě srdečního stahu. Touto periodou poté budeme segmentovat echokardiografický záznam podle algoritmu v sekci 3.7.

Diskrétní Fourierova (DFT) transformace je odvozena z Fourierovy transformace (FT) pro spojitý signál [18]. Idea obou těchto transformací vychází z Fourierových řad. Ty dovolují periodický signál vyjádřit jako součet jeho harmonických složek. Tyto harmonické složky

popisujeme pomocí komplexních exponenciál (goniometrických funkcí). Také neperiodické funkce můžeme vyjádřit součtem jejich harmonických složek. V tomto případě využijeme Fourierovy transformace, které stejně jako Fourierovy řady převedou funkci času $x(t)$ do její frekvenční domény. Vzorec Fourierovy transformace pro spojitý signál je následující:

$$X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt \quad (3.27)$$

Výsledné $X(\omega)$ představuje frekvenční spektrum signálu, které je komplexní. Symbol $\omega = 2\pi f$ představuje kruhovou frekvenci. Výraz $e^{-j\omega t}$ je exponenciální tvar komplexního čísla, jehož obor hodnot leží na jednotkové kružnici. Funkce je definována pro všechny frekvence od $-\infty$ do ∞ . Pro většinu signálů nám stačí znát pouze pravou část spektra ($\omega > 0$), protože část levá je sní komplexně sdružená [18]. Tato transformace je reverzibilní a proto můžeme signál ze spektra znovu restaurovat pomocí vzorce ve tvaru:

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega)e^{j\omega t} d\omega \quad (3.28)$$

Vstupem Fourierovy transformace je spojitá funkce času. V našem případě však pracujeme s diskretními hodnotami, které odpovídají průměrnému pohybu bodů v každém snímku vzhledem k prvnímu snímku echokardiografického záznamu (rovn. 3.26). Každý vzorek signálu tedy odpovídá jednomu snímku záznamu. Aplikaci Fourierovy transformace na diskretní signál provedeme náhradou integrálu za sumu [11]:

$$X(\omega) = \sum_{n=-\infty}^{n=\infty} x[n]e^{-j\omega t} dt \quad (3.29)$$

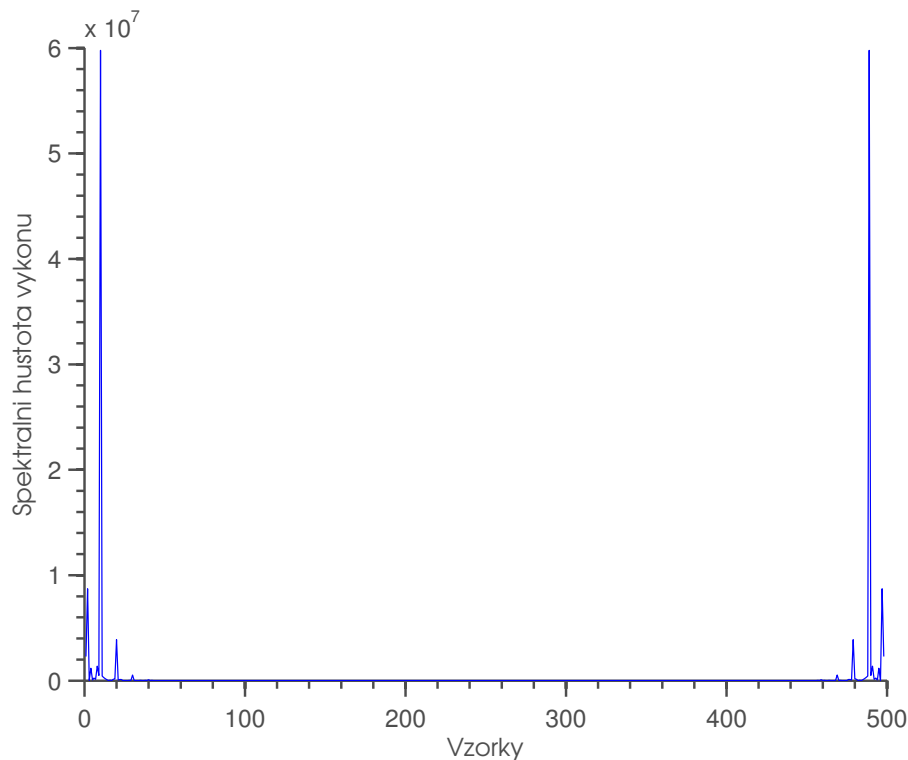
Tato transformace se nazývá Fourierova transformace s diskretním časem (DTFT). Všimněte si, že na rozdíl od první formule (rovn. 3.27) můžeme $X(\omega)$ spočítat, protože v praktických aplikacích nemáme nekonečně mnoho vzorků. Pro výpočty na počítači ovšem stále zůstává problém nekonečné délky vypočteného spektra. Tento problém řeší diskretní Fourierova transformace:

$$X[k] = \sum_{n=0}^{N-1} x[n]e^{-j\frac{2\pi}{N}kn} \quad (3.30)$$

Diskretní Fourierova transformace dále vzorkuje výsledné spektrum DTFT a tak pro N vstupních vzorků obdržíme N vzorků spektra [17]. Tyto vzorky považujeme za periodické, neboť pracujeme s periodickými prodlouženími signálů ze základního intervalu. Pro syntézu původního signálu použijeme inverzi této transformace ve tvaru:

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k]e^{j\frac{2\pi}{N}kn} \quad (3.31)$$

Výpočet diskretní Fourierovy transformace podle vzorců 3.30 a 3.31 vyžaduje $O(n^2)$ operací sčítání nebo násobení s komplexními čísly [11]. Efektivní algoritmy pro výpočet diskretní Fourierovy transformace se nazývají rychlá Fourierova transformace. Původní algoritmus popsali Cooley a Tukey v roce 1965 a je navržen pro délku transformace rovnou



Obrázek 3.9: Spektrální hustota výkonu signálu z obrázku 3.8.

mocninám dvěma [5]. Tento algoritmus využívá periodičnosti a symetrií komplexní exponenciály [11]. Dnes existuje mnoho algoritmů rychlé Fourierovy transformace a používají se nejen ke spektrální analýze signálů, ale i výpočtu konvoluce, frekvenčních charakteristik a číslicovou filtraci.

Ještě před aplikací rychlé Fourierovy transformace je nutno signál předzpracovat odstraněním jeho stejnosměrné složky. Stejnosměrná složka signálu získaného analýzou pohybu srdeční stěny je pouze dodatek a nenes tak pro nás žádnou relevantní informaci. Vzniká při nepřesnostech výpočtu optického toku. Její odstranění provedeme odečtením průměrné hodnoty signálu:

$$signal = signal - mean(signal) \quad (3.32)$$

Protože signál s nímž pracujeme má náhodný charakter (nemůžeme deterministicky určit jak bude vypadat jeho průběh), budeme jeho spektrum analyzovat pomocí spektrální hustoty výkonu (obr. 3.9) [18]. Tato veličina udává rozložení výkonu ve frekvenční oblasti. Z tohoto rozložení můžeme určit, které frekvence jsou v našem signálu dominantní. Spektrální hustotu výkonu frekvenčních složek vypočítáme pomocí následujícího vzorce:

$$PSD = |X(\omega)|^2 \quad (3.33)$$

Poslední komponenta, která nám schází k výpočtu periody dominantní frekvence signálu je frekvenční osa. Aplikací rychlé Fourierovy transformace obdržíme frekvenční spektrum s N vzorky. Nás zajímá na jakých frekvencích dané vzorky leží. Víme, že signál je vzorkován po jednom snímku a podle Shannonova vzorkovacího teorému by měl mezi frekvencí signálu f a vzorkovací frekvencí f_s platit následující vztah:

$$f_s > 2 \cdot f \quad (3.34)$$

Ze vztahu 3.34 určíme hraniční vzorkovací frekvenci jejíž hodnota bude $\frac{1}{2}$. Tato frekvence se v literatuře označuje jako Nyquistova a náleží vzorku, jenž leží ihned za posledním vzorkem frekvenčního spektra [17]. Vlastností spektra signálu tvořeného reálnými čísly je, že první polovina je komplexně sdružená s druhou polovinou viz obrázek 3.9. Tuto vlastnost lze odvodit ze vzorce diskrétní Fourierovy transformace (rovn. 3.30)[11]. Využijeme-li těchto poznatků, můžeme k jednotlivým vzorkům frekvenčního spektra přiřadit frekvence pomocí následující rovnice:

$$freq[i] = \frac{i}{N} \cdot \frac{1}{2}, \quad i \in \left(0, \frac{N}{2}\right) \quad (3.35)$$

Nyní již stačí najít frekvenci s největší spektrální hustotou výkonu f_{dom} :

$$f_{dom} = \max(PSD) \quad (3.36)$$

A periodu této dominantní frekvence spočítáme pomocí následujícího vzorce:

$$T = \frac{1}{f_{dom}} \quad (3.37)$$

Spočtená perioda bude přibližně odpovídat jedné periodě srdečního stahu. Jelikož každý srdeční cyklus nemusí trvat stejnou dobu, musíme implementovat dodatečné upřesnění při hledání hranic jednotlivých segmentů. Tento problém a celkové shrnutí algoritmu segmentace echokardiografického záznamu je v následující sekci 3.7.

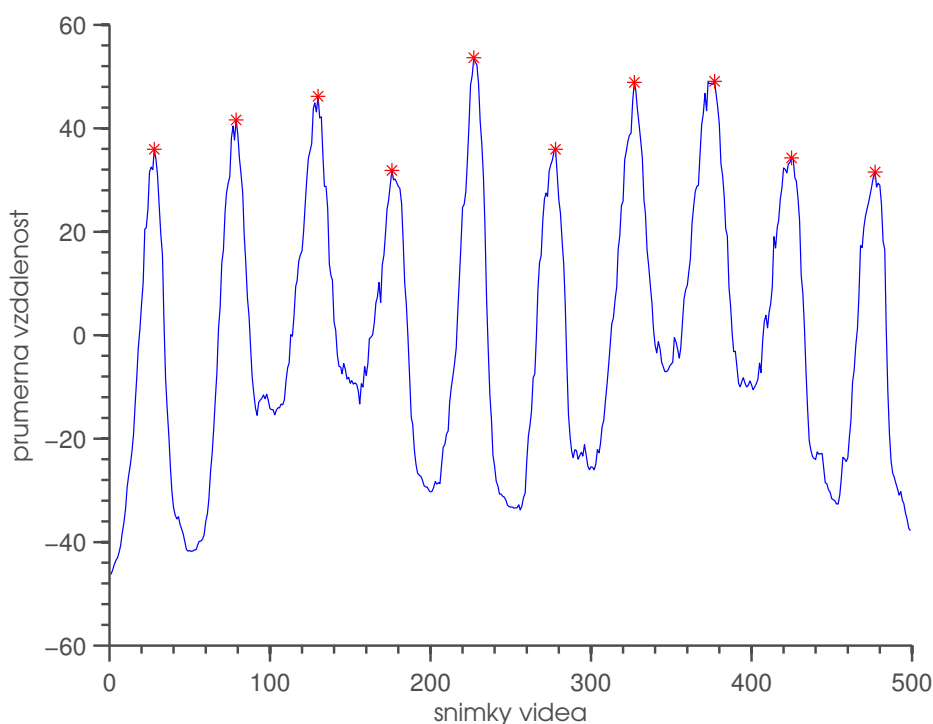
3.7 Algoritmus segmentace echokardiografického záznamu

Tato kapitola shrnuje teoretické poznatky z předcházejících sekcí a uvádí kompletní popis algoritmu pro časovou segmentaci echokardiografického záznamu (alg. 3). Jednotlivé segmenty budou mít rozsah od počátku vypuzovací fáze srdce po začátek následující vypuzovací fáze viz kapitola 2. Vstupem algoritmu je video nesoucí v kruhovém výřezu ultrazvukový záznam pohybu srdce z echokardiografického vyšetření (obr. 3.5) a parametry kruhové výseče. Výstupem jsou rozsahy snímků jednotlivých segmentů.

Algoritmus nejprve načte parametry kruhové výseče a určí minimální obdelníkovou oblast, ve které se nachází kruhová výseč s ultrazvukovým záznamem. Poté tuto oblast pokryje čtvercovou mřížkou bodů, jenž budou odpovídat sledovaným oblastem algoritmu Lucas-Kanade. Dále tuto mřížku redukuje pouze na body, které leží uvnitř kruhové výseče. Nyní máme určené oblasti, které budou sledovány algoritmem Lucas-Kanade. Detailněji jsou tyto kroky popsány v sekci 3.5.

Následně načítáme snímky videa a ořezáváme je obdelníkovým oknem na oblast obsahující pouze kruhovou výseč. Poté aplikujeme algoritmus Lucas-Kanade (alg. 1) tím způsobem, že vždy hledáme umístění bodů na aktuálním snímku vzhledem k prvnímu snímku videa. Z nově nalezených pozic sledovaných bodů vypočítáme signál jako průměrný pohyb všech bodů mezi oběma snímky videa (rovn. 3.26).

Dalším krokem je rozklad signálu do jeho frekvenčních složek. Tuto transformaci provedeme pomocí diskrétní Fourierovy transformace (rovn. 3.30). Jelikož náš signál je náhodný provedeme analýzu spektra pomocí spektrální hustoty výkonu (rovn. 3.33). Následně ve



Obrázek 3.10: Vyznačení segmentů echokardiografického záznamu.

frekvenčním spektru nalezneme dominantní frekvenci (frekvence s největší hustotou výkonu). Poté vypočítáme periodu této frekvence.

Posledním částí programu je algoritmus segmentace echokardiografického záznamu. Segmentace by měla začínat z bodu, kdy začíná nebo končí systolická fáze. Toto zajistíme tak, že budeme segmentovat od maximální hodnoty signálu. V tomto bodě se srdeční sval přestane napínat a začne docházet k jeho relaxaci. Segmentaci tedy zahájíme od bodu určeného jako:

$$seg_{start} = \max(signal) \quad (3.38)$$

Jelikož je doba srdečních cyklů proměnná (vrcholy signálu neleží přesně na násobcích periody od počátku segmentace), je nutno hranice jednotlivých segmentů zpřesnit [20]. Nestačí nám tedy pouze krokovat po periodě dominantní frekvence, ale musíme v okolí každého kroku dodatečně hledat maximum signálu. Toto maximum nám stejně jako v předchozím případě hledání startovního bodu segmentace představuje konec jedné srdeční fáze. Velikost této oblasti jsme určily jako dvacet procent periody dominantní frekvence na obě strany kroku. Hledání vrcholů signálu je znázorněno algoritmem 2. Tento algoritmus aplikujeme dvakrát. Pro hledání vrcholů napravo od počátku segmentace s kladnou periodou, pro hledání vrcholů nalevo se zápornou periodou.

Nyní již máme určeny hranice jednotlivých segmentů. Je-li to nutné, tak k nim doplníme ještě hranice začátku a konce videa. Zobrazení jednotlivých segmentů signálu obdrženého analýzou pohybu srdeční stěny metodou Lucas-Kanade je na obrázku 3.10. Kompletní přepis algoritmu segmentace echokardiografického záznamu je uveden jako algoritmus 3.

Algoritmus 2: Nalezení hranic jednotlivých segmentů.

Cíl: Krokuj signálem po periodě z dané počáteční pozice.

Vstup: Signál $signal$, počáteční bod seg_{start} , perioda T .

$explore = 1$

$index = seg_{start}$

$oblast = \frac{T}{100} \cdot 20$

while ($explore == 1$) **do**

$explore = 0$

$index = index + T$

if ($(index \geq 0) \wedge (index < length(signal))$) **then**

$explore = 1$

if ($index - oblast < 0$) **then**

$index = max_{index}(signal(0 : index + oblast))$

else if ($index + oblast \geq length(signal)$) **then**

$index = max_{index}(signal(index - oblast : length(signal) - 1))$

else

$index = max_{index}(signal(index - oblast : index + oblast))$

end

$segments.push(index)$

end

end

Řešení: Hranice jednotlivých segmentů jsou v proměnné $segments$.

Algoritmus 3: Časová segmentace echokardiografického záznamu.

Cíl: Proved č časovou segmentaci echokardiografického záznamu.

Načti souřadnice středu, úhel a poloměr kruhové výseče: S, α, r .

Zpočítej obdelníkový výřez: $vyrez = [(S_x - a, S_y), (S_x + a, S_y + r)]$. (rovn. 3.25)

Vyber body podle rovnice: $bod \in B \iff (\vec{V} \cdot \vec{L}_n \geq 0) \wedge (\vec{V} \cdot \vec{P}_n \geq 0)$. (rovn. 3.24)

Načti první snímek echokardiografického záznamu F_1 .

for $i = 1$ **to** $pocetSnimku$ **do**

Aplikuj pyramidový algoritmus Lucas-Kanade: $pyrLK(F_1, F_i, B, B_{lk})$. (alg. 1)

Spočítej položku signálu: $signal[i] = \frac{\sum_{k=0}^{k=N} |B[k] - B_{lk}[k]|}{N}$. (rovn. 3.26)

end

Odstraň stejnosměrnou složku signálu: $signal = signal - mean(signal)$ (rovn. 3.32)

Vypočítej frekvenční spektrum X : $X[k] = \sum_{n=0}^{N-1} signal[n]e^{-j\frac{2\pi}{N}kn}$ (rovn. 3.30)

Vypočítej spektrální hustotu výkonu PSD : $PSD[k] = |X[k]|^2$ (rovn. 3.33)

Najdi dominantní frekvenci f_{dom} : $f_{dom} = max(PSD)$ (rovn. 3.36)

Urči periodu dominantní frekvence T : $T = \frac{1}{f_{dom}}$ (rovn. 3.37)

Nalezni startovní bod segmentace seg_{start} : $seg_{start} = max(signal)$ (rovn. 3.38)

Urči velikost oblasti $oblast$ pro upřesnění hranic: $oblast = \frac{T}{100} \cdot 20$

Urči hranice segmentů $segments$ s kladným a záporným T . (alg. 2)

Doplň $segments$ o první a poslední snímek echokardiografického záznamu.

Řešení: Hranice jednotlivých segmentů jsou v proměnné $segments$.

Kapitola 4

Implementace segmentace echokardiografického záznamu.

Program pro segmentaci echokardiografického záznamu je implementován v jazyce C++. Výhodou zvoleného jazyka pro naši aplikaci je především volná dostupnost knihoven pro zpracování obrazu a analýzu signálu. Dále v této aplikaci klademe důraz na rychlost programu, neboť analýza pohybu bodů metodou Lucas-Kanade je výpočetně náročný úkol. Výsledný program je možno spustit pod systémy Linux a Windows. Pro některé části tohoto programu byl nejprve vytvořen prototyp v prostředí Matlab. Díky tomu jsme mohli teoretické poznatky jednoduše převést do programového prostředí a ověřit si jejich platnost v naší aplikaci.

Pro analýzu pohybu srdeční stěny v echokardiografickém záznamu jsme využily knihovny OpenCV. Tato populární knihovna je volně dostupná¹ a obsahuje mnoho funkcí z oblasti počítačového vidění. Seznam, použití a popis mnoha funkcí z této knihovny je možno nalézt např. v [3]. OpenCV obsahuje několik funkcí optického toku, mezi nimiž lze nalézt i implementaci metody optického toku Lucas-Kanade nad pyramidou snímků.

Získání frekvenčních složek z vypočteného signálu jsme provedli rychlou Fourierovou transformací implementovanou v knihovně FFTW². Tato knihovna je volně dostupná a umožňuje jedno i vícedimensionální transformace nad komplexním i reálným vstupem. Více o této knihovně, která se umísťuje na předních místech v mnoha benchmarcích, je možno nalézt například v [8].

4.1 Implementace metody Lucas-Kanade

Knihovna OpenCV přichází se dvěma metodami výpočtu optického toku. První variantou je metoda Lucas-Kanade, která je popsána v sekci 3.2. OpenCV pro tuto metodu podporuje výpočet nad pyramidou snímků, jenž je implementován funkcí `calcOpticalFlowPyrLK()`. Druhá implementovaná metoda se nazývá Horn-Schunckova. Tato metoda je narozdíl od metody Lukas-Kanade globální a Aperture problem (sekce 3.1) řeší výpočtem vektoru pohybu pro všechny pixely snímku [12]. V naší aplikaci je použita implementace metody Lucas-Kanade, neboť je rychlejší a pro Horn-Schunckovu metodu chybí podpora výpočtu nad pyramidou snímků. Definice vybrané funkce `calcOpticalFlowPyrLK()` je následující:

¹<http://opencv.org>

²<http://www.fftw.org>

```

void calcOpticalFlowPyrLK(InputArray prevImg,
    InputArray nextImg,
    InputArray prevPts,
    CV_OUT InputOutputArray nextPts,
    OutputArray status,
    OutputArray err,
    Size winSize=Size(21,21),
    int maxLevel=3,
    TermCriteria criteria=TermCriteria(TermCriteria::COUNT+
        TermCriteria::EPS, 30, 0.01),
    int flags=0,
    double minEigThreshold=1e-4);

```

První dva parametry funkce `calcOpticalFlowPyrLK()` představují původní a následující snímek videa. Funkce předpokládá, že se jedná o jedno kanálové obrázky ve stupních šedi. Tato transformace je v našem programu provedena pomocí funkce `cvtColor()` s parametrem `code` rovným konstantě `CV_BGR2GRAY`:

```

|| void cvtColor(InputArray src, OutputArray dst, int code, int dstCn=0);

```

Parametr funkce `calcOpticalFlowPyrLK()` `prevPts` představuje souřadnice pozic bodů v kruhové výseči echokardiografického záznamu (sekce 3.5). Souřadnice jejich pozic na následujícím snímku videa jsou ukládány do proměnné, která je zároveň parametrem `nextPts`. Parametry `status` a `error` určují, zda-li se jednotlivé body podařilo nalézt, případně jakou chybou skončil výpočet. Nastavení vhodného počtu úrovní pyramidy snímků `maxLevel` a velikosti okna `winSize` pro naše účely bylo provedeno testováním popsáním v sekci 4.2. Implementace optického toku je v naší aplikaci řešena následovně:

```

/* ... */

if (!capt_rf->read(frame))
{
    cout << "ERROR: Can't read a video file." << endl;
    return -1;
}
frame = frame(region_OI);
cvtColor(frame, gr_pr_frame, CV_BGR2GRAY);

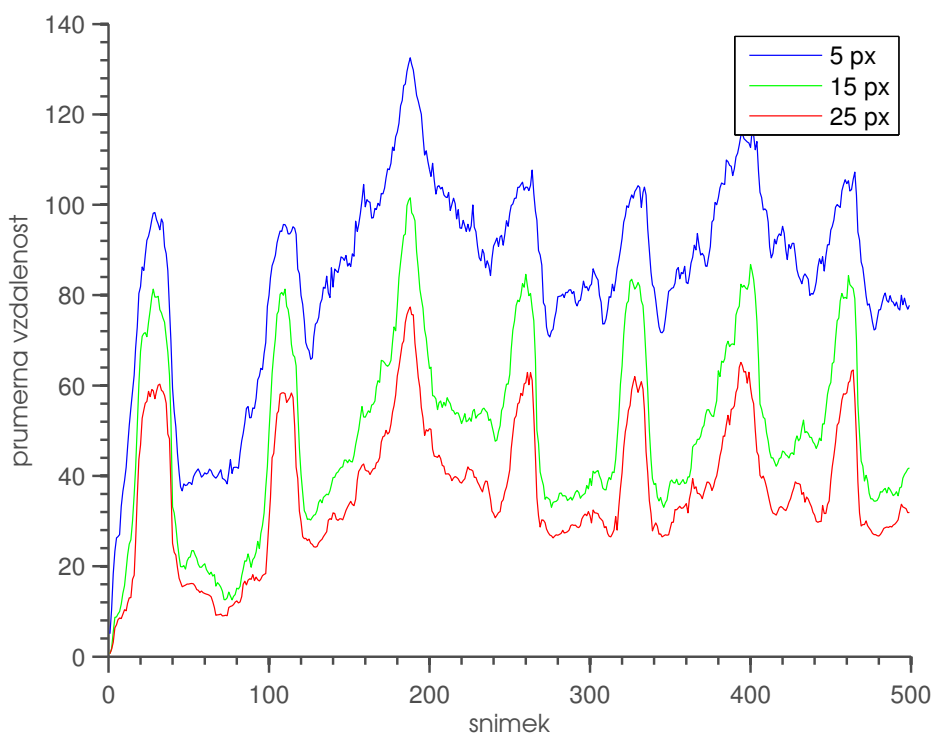
/* ... */

while (1)
{
    if (!capt_rf->read(frame))
        break;
    frame = frame(region_OI);
    cvtColor(frame, gr_frame, CV_BGR2GRAY);
    calcOpticalFlowPyrLK(gr_pr_frame, gr_frame, features, next_features,
        status, error);

    /* ... */
}

/* ... */

```

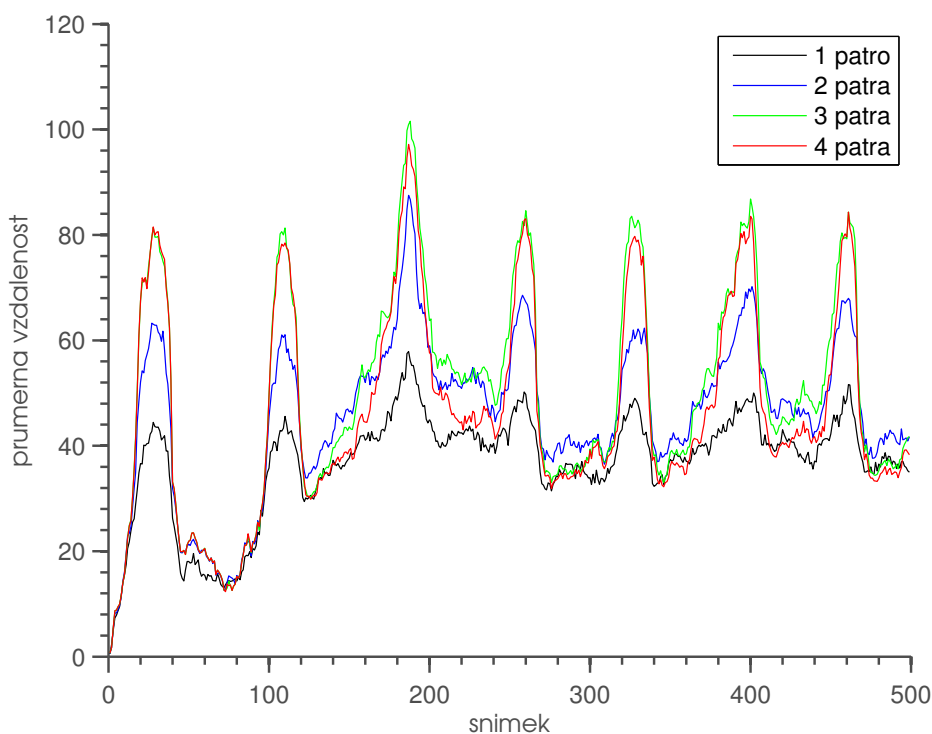
Obrázek 4.1: Vliv velikosti okna optického toku na výsledný signál.

4.2 Testování parametrů optického toku

Pro co nejpřesnější odhad segmentů srdečních stahů v echokardiografickém záznamu, bylo nutné provést kalibraci výpočtu optického toku. Tento výpočet je realizován funkcí z knihovny OpenCV `calcOpticalFlowPyrLK()` (sekce 4.1). Kalibrace této funkce probíhala testováním vhodnosti velikosti okna optického toku (parametr `winSize`) a počtu úrovní pyramid snímků (parametr `maxLevel`). Další parametry této funkce jsme ponechaly ve výchozím nastavení.

Metodologie testování byla následující: Nejprve jsme náhodně zvolily 3 echokardiografické záznamy. Na těchto záznamech jsme prvně počítaly optický tok s různými velikostmi okna. Počet pater pyramid byl při tomto testování ponechán na hodnotě tři, protože tato hodnota je vzhledem k rozlišení videa dostatečná [2]. Měnily jsme tedy pouze hodnoty parametru `winSize`. Rozmezí testovaných hodnot pro hrany čtvercového okna optického toku bylo od 5 do 30 pixelů. Kritériem pro výběr vhodného nastavení byl průběh signálu podle rovnice 3.26. Čím lépe byla ze signálu čitelná dominantní frekvence (sekce 3.6), tím bylo nastavení velikosti okna lepší.

Výsledky testování velikosti okna z jednoho záznamu jsou zobrazeny na obrázku 4.1. Při všech třech nastaveních je jasně patrná dominantní frekvence. Pro velikost okna 5 pixelů je vidět, že signál obsahuje největší stejnosměrnou složku. Také je více kostrbatý, což můžeme vzhledem k charakteru pohybu srdce považovat za nežádoucí. Zbylé dvě nastavení 15 a 25 pixelů jsou velice podobné, ale dominantní frekvence je více patrná pro velikost okna 15 pixelů. U zbylých dvou testů jsem došel ke schodným závěrům. Velikost okénka optického toku je tedy nastavena na 15 pixelů.



Obrázek 4.2: Vliv počtu pater pyramid snímků na výsledný signál.

Další testování proběhlo nad počtem úrovní pyramid snímků. Použitá metodologie se schoduje s metodologií při kalibraci velikosti okénka optického toku. Velikost okénka byla nastavena na 15 pixelů. Rozsah testovaných pater pyramid snímků se pohyboval v rozmezí od jednoho patra až po čtyři patra. Výsledky testování z jednoho echokardiografického záznamu jsou zobrazeny na obrázku 4.2.

Z výsledků na obrázku 4.2 vyplývá, že pouze jediné patro je pro náš výpočet nevyhovující. Vidíme, že signál vypočtený s jedním i dvěma patry pyramid nedosahuje v průběhu takových změn. To je způsobeno použitím optického toku, kde hledáme umístění bodů všech snímků vzhledem k prvnímu snímku videa. Použitím takové pyramid tak nelze nalézt umístění oblastí, které jsou při periodickém srdečním pohybu dále vzdálené. Rozdíly mezi třemi a čtyřmi patry je minimální. Pro segmentaci jsme tedy zvolili tři patra. Pro přehlednost textu jsou výsledky zbylých testů umístěny v příloze A.

4.3 Implementace diskrétní Fourierovy transformace

Výpočet frekvenčního spektra je proveden pomocí rychlé Fourierovy transformace implementované v knihovně FFTW [8]. Pro realizaci tohoto výpočtu je nutné použití několika funkcí z této knihovny, které jsou uvedeny dále. Prvním krokem je alokace vstupně výstupních dat. Data pro zpracování knihovnou FFTW se musí alokovat a dealokovat pomocí funkcí `fftw_malloc` a `fftw_free`. V našem případě již máme vstupní signál připraven a tak alokujeme pouze výstupní data. Definice funkcí `fftw_malloc` a `fftw_free` je následující:

```
|| void *fftw_malloc(size_t n);
|| void fftw_free(void *p);
```

Dalším krokem je vytvoření plánu transformace. Plánem se v knihovně FFTW rozumí popis vstupních i výstupních dat, určení typu transformace a jakým způsobem se provede výpočet rychlé Fourierovy transformace. Po dokončení transformace je nutno tento plán dealokovat.

```

|||  fftw_plan fftw_plan_r2r_1d(int n, double *in, double *out,
|||                               fftw_r2r_kind kind, unsigned flags);
|||  void fftw_destroy_plan(fftw_plan plan);

```

Vstupní i výstupní data jsou v naší aplikaci tvořeny reálnými hodnotami. Transformace je jednodimenzionální a pro postup výpočtu je zvolena nejrychlejší varianta heuristikou pomocí konstanty `FFTW_ESTIMATE`. Po definici plánu můžeme provést samotný výpočet frekvenčního spektra. Výpočet zahájíme následující funkcí:

```

|||  void fftw_execute(const fftw_plan plan);

```

Shrneme-li získané poznatky pro výpočet frekvenčního spektra knihovnou FFTW, tak můžeme tento výpočet realizovat následujícím kódem:

```

|||  /* ... */
|||
|||  ft_input = &signal[0];
|||  ft_output = (double *) fftw_malloc(sizeof(double)*N);
|||  ft_plan = fftw_plan_r2r_1d(signal.size(), ft_input, ft_output, FFTW_R2HC
|||                          , FFTW_ESTIMATE);
|||  fftw_execute(ft_plan);
|||
|||  /* ... */
|||
|||  fftw_destroy_plan(ft_plan);
|||  fftw_free(ft_output);
|||
|||  /* ... */

```

4.4 Implementace rekonstrukce EKG křivky

Následující kapitola 5 se zabývá testování přesnosti vypočtených segmentů echokardiografického záznamu. Pro toto testování byla využita EKG křivka, na které jsme určovali polohy hranic jednotlivých segmentů. EKG křivka je zobrazena spolu s posuvníkem v dolní části všech echokardiografických záznamů (obr. 3.5). Její rekonstrukce proběhla sledováním polohy posuvníku, jenž zobrazuje aktuální polohu v čase na této křivce.

Implementace rekonstrukce EKG křivky sledováním polohy posuvníku byla provedena pomocí knihovny OpenCV. Redukce informace ve videu na pouhé zobrazení posuvníku využívá filtraci na základě barvy, neboť tento posuvník je jedinou červenou entitou ve videu. Proto je vhodné jako první krok převést obraz z aditivního barevného modelu RGB na model HSV. V tomto barevném modelu lze snadněji určit hranice barev, protože odděluje informace o barvě a intenzitě [4]. Převod snímků z RGB na HSV model byl proveden pomocí funkce `cvtColor()` definované v sekci 4.1 s parametrem `CV_BGR2HSV`.

Interval hodnot odpovídající červené barvě lze určit z válcového souřadnicového systému modelu HSV. Tento souřadnicový systém je běžnou součástí grafických softwarů. Červená barva leží v tomto souřadnicovém systému na počátku i na konci a je tak definována dvěma intervaly. Dalším krokem je filtrace obrazu, během které se obraz převede do binární podoby. Pixely, jejichž barva odpovídala červené nesou hodnoty jedna, ostatní pixely nesou hodnotu

nula. Tuto filtraci lze provést pomocí funkce z knihovny OpenCV `inRange()` definované jako:

```
|| void inRange(InputArray src, InputArray lowerb,  
||             InputArray upperb, OutputArray dst);
```

Jelikož červenou barvu definují v modelu HSV dva intervaly je nutné aplikovat funkci `inRange()` dvakrát. Výsledek poté obdržíme provedením operace OR nad oběma výslednými obrázky. Operace OR nad maticemi je implementována v OpenCV pomocí funkce `bitwise_or()`. Následující kód implementuje transformace snímků videa do binární podoby obsahující pouze původní červené entity:

```
|| /* ... */  
||  
|| while (capt_rf.read(frame))  
|| {  
||     cvtColor(frame, frame_hsv, CV_BGR2HSV);  
||     inRange(frame_hsv, Scalar(0, 100, 100), Scalar(10, 255, 255),  
||             frame_t);  
||     inRange(frame_hsv, Scalar(160, 100, 100), Scalar(179, 255, 255),  
||             frame_t1);  
||     bitwise_or(frame_t, frame_t1, frame_t);  
||  
||     /* ... */  
|| }  
||  
|| /* ... */
```

Rekonstrukce EKG křivky představuje sledování y-ové souřadnice posuvníku ve videu. Toto sledování provedeme deterministickým nalezením prvního bodu posuvníku a přečtením jeho y-ové souřadnice. Jelikož je počáteční bod snímku definován vlevo nahoře je nutné pro rekonstrukci EKG křivky sledovanou souřadnici odečíst od dostatečně velké konstanty. Pro hledání polohy složitějších objektů je vhodné použít momenty obrázku [13].

Kapitola 5

Měření přesnosti algoritmu segmentace

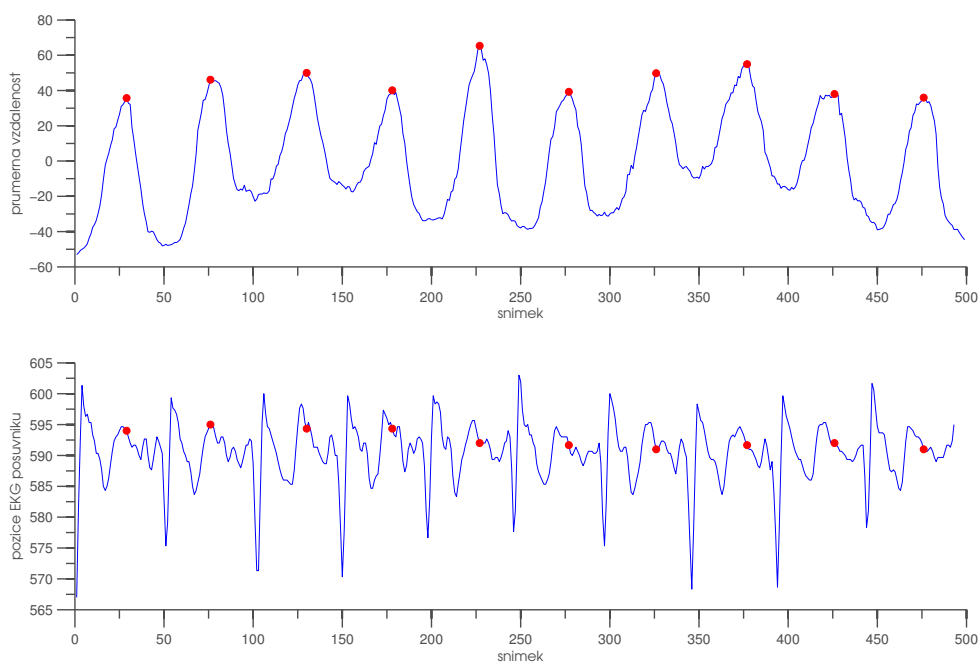
Pro zjištění, jak přesně náš algoritmus segmentuje echokardiografický záznam, jsme využili možnosti porovnat polohy hranic jednotlivých segmentů s jejich umístěním na EKG křivce. Ke všem testovaným echokardiografickým záznamům je přiložen soubor ve formátu h5, který zobrazuje elektrickou aktivitu srdce v průběhu echokardiografického vyšetření. Bohužel u těchto souborů je obtížné definovat vztah jak namapovat EKG křivku na snímky videa. Proto jsme EKG křivku vždy rekonstruovali z echokardiografických záznamů. Implementaci této segmentace se zabývá sekce 4.4.

Pro naše měření jsme využili osmi dostupných echokardiografických záznamů. V průběhu každého měření nejprve určíme jak vypadá segmentovaný signál. Pokud tvar signálu neodpovídá očekávanému pohybu srdeční stěny (obr. 3.8), tak měření považujeme za neúspěšné. Dalším krokem je kontrola umístění hranic jednotlivých segmentů. Tento krok závisí především na spočtené dominantní frekvenci (rovn. 3.36). Signál může obsahovat výkyvy a v tom případě budou hranice segmentů spočtené z dominantní frekvence špatně určeny. Poslední krokem je určení v jakých srdečních fázích se hranice segmentů nachází. Ideálně by náš algoritmus měl vždy segmentovat podle stejné srdeční fáze. Pro tuto kontrolu využijeme rekonstruovanou EKG křivku, na kterou umístíme hranice segmentů. Z poloh těchto hranic na EKG křivce poté určíme srdeční fáze. Touto metodikou ověříme, že jednotlivé segmenty jsou správně zarovnány a odpovídají periodickému pohybu srdeční stěny.

5.1 Měření přesnosti jednotlivých záznamů

Ke každému provedenému testu je přiřazen obrázek se dvěma grafy. Horní graf zobrazuje segmentovaný signál, který jsme obdrželi aplikací metody Lucas-Kanade na oblast srdeční stěny v echokardiografickém záznamu. Dolní graf zobrazuje rekonstruovanou EKG křivku ze stejného záznamu. Červené body na obou grafech značí polohy hranic jednotlivých segmentů. Segment je tedy vždy určen svou levou a pravou hranicí případně prvním či posledním snímkem echokardiografického záznamu. Tento obrázek poté analyzujeme metodologií popsanou výše v této kapitole.

Segmentace prvního signálu s přiloženou EKG křivkou je zobrazena na obrázku 5.1. Kvalita tohoto záznamu je velice dobrá. Srdeční stěna je zde výrazná, pohyb chlopní nevýrazný a výskyt šumu je minimální. Tvar signálu tak odpovídá očekávanému periodickému pohybu srdeční stěny. Průměrná vzdálenost se periodicky zvyšuje a snižuje s tím, jak se



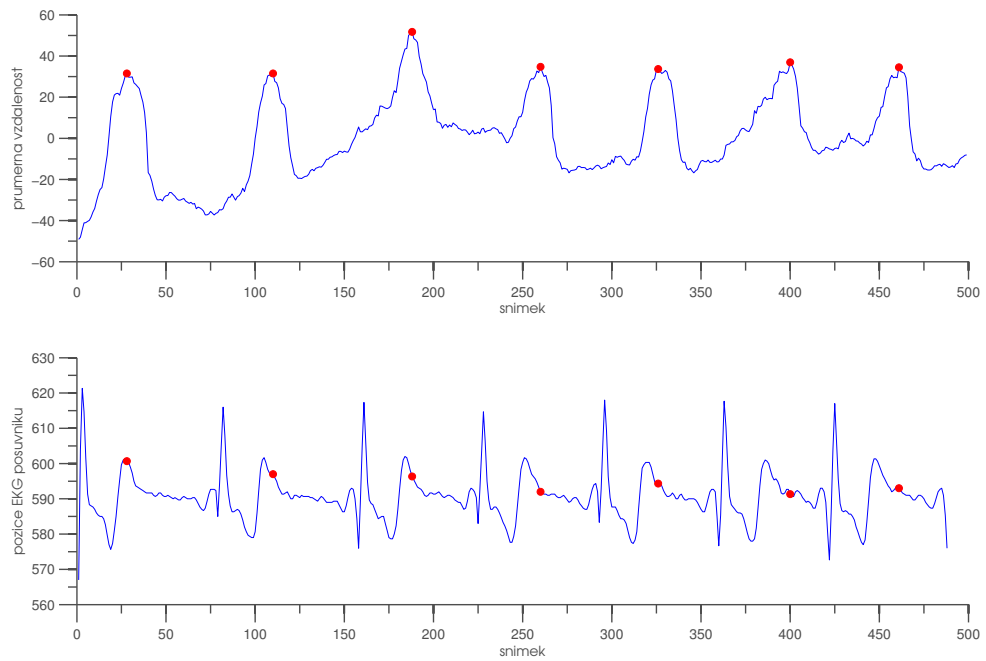
Obrázek 5.1: Signál a polohy hranic segmentů prvního echokardiografického záznamu.

střídá systolická a diastolická srdeční fáze. Jednotlivé segmenty jsou také správně určeny neboť jim odpovídají rozsahy mezi dvěma kladnými hroty signálu. Hranice segmentů na příslušné EKG křivce leží v úseku ST (sekce 2.6). V tomto úseku je srdce ve fázi plateau a nevykazuje žádnou elektrickou aktivitu. V této fázi je srdeční sval nejvíce napnutý a postupně nastává fáze diastoly.

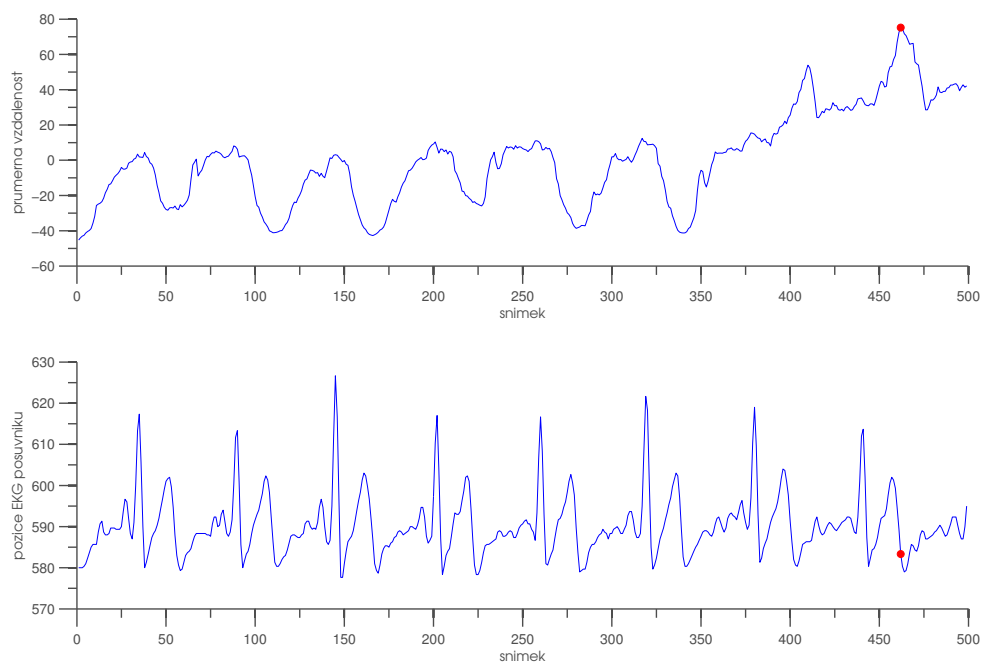
Segmentace druhého signálu se svou rekonstruovanou EKG křivkou je zobrazena na obrázku 5.2. Kvalita druhého záznamu je dobrá. Pohyb srdeční stěny a mezikomorové chlopně je zde výrazný a záznam obsahuje minimum šumu. Tvar tohoto signálu stejně jako v prvním případě odpovídá očekávanému periodickému pohybu srdeční stěny. I zde se průměrná vzdálenost mění podle srdeční fáze (sekce 2.1). Polohy jednotlivých segmentů jsou také správné, neboť leží mezi kladnými hroty signálu. Projekce hranic jednotlivých segmentů na EKG křivku nám stejně jako v případě prvního testu vždy dává úsek ST.

Segmentace třetího snímku s příslušnou EKG křivkou je zobrazena na obrázku 5.3. Kvalita tohoto záznamu je špatná. Třetí záznam obsahuje špatně rozeznatelnou srdeční stěnu, která se v průběhu záznamu stále více ztrácí. Ze signálu třetí segmentace lze opět vyčíst, že správně sleduje periodický pohyb srdeční stěny. Nicméně zhruba v poslední třetině tohoto signálu se nachází značné množství šumu. Tento šum je způsoben špatnou kvalitou echokardiografického záznamu. Srdeční stěna je v poslední třetině špatně čitelná i pro člověka. Proto je tento záznam nevhodný pro analýzu metodami optického toku. Dále vidíme, že algoritmus určí dominantní frekvenci vzhledem k nejvyššímu kladnému hrotu signálu. Tento hrot však leží již v zašumělé oblasti a tedy segmentace tohoto echokardiografického záznamu je neúspěšná.

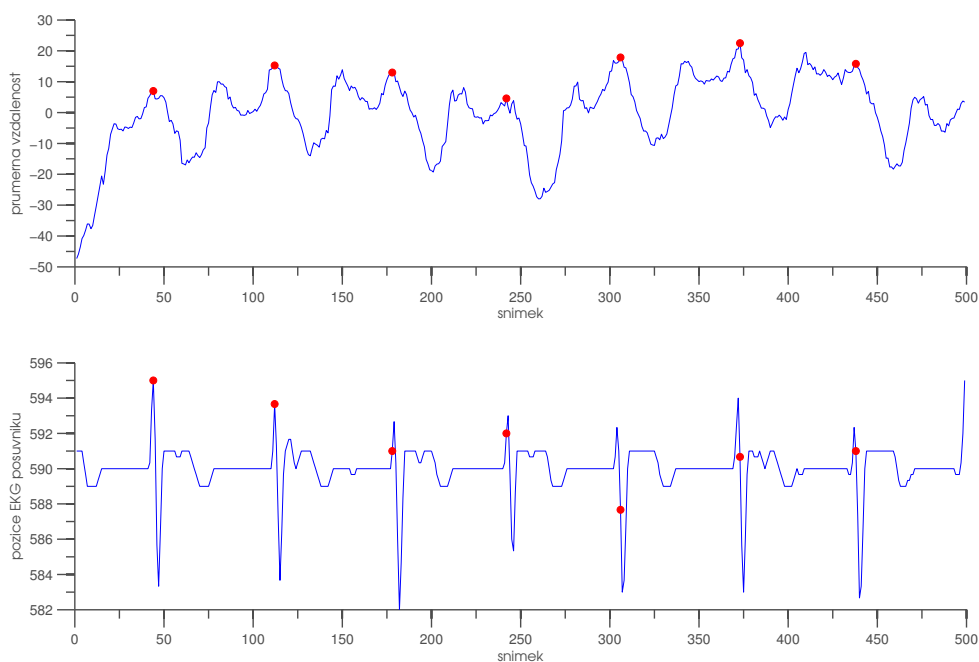
Výsledky segmentace čtvrtého echokardiografického záznamu jsou zobrazeny na obrázku 5.4. Kvalita záznamu je velice dobrá. Pohyb srdeční stěny je snadno čitelný, chlopeň je v tomto záznamu velice výrazná a šum je minimální. Signál je opět periodický,



Obrázek 5.2: Signál a polohy hranic segmentů druhého echokardiografického záznamu.



Obrázek 5.3: Signál a polohy hranic segmentů třetího echokardiografického záznamu.

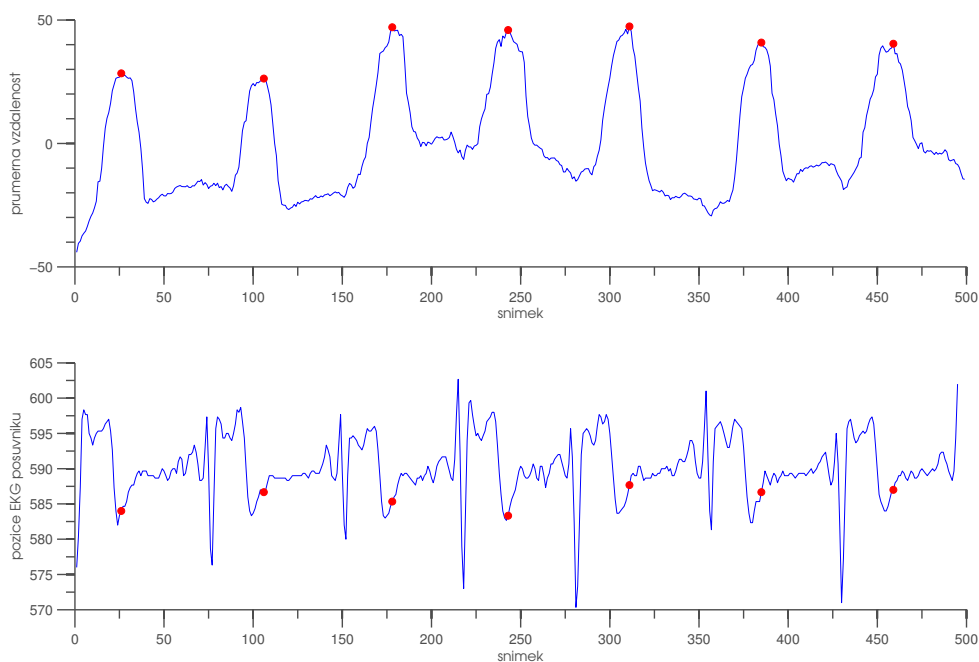


Obrázek 5.4: Signál a polohy hranic segmentů čtvrtého echokardiografického záznamu.

nicméně místo jediného kladného hrotu signálu leží blízko sebe hroty dva. Segmenty jsou opět správně určeny a leží vždy mezi pravými hroty. Projekce hranic těchto hrotů na EKG křivku nám vždy dává QRS interval. V tomto intervalu nastává systola komor. Jedná se tedy o segmentaci v jiné fázi, než ve které byly nasegmentovány první a druhý záznam. Všimněme si, že kdybychom vždy segmentovali levý roh, tak opět obdržíme úsek ST. Vznik druhého hrotu v signálu je způsoben sledováním pohybu chlopně. Tento pohyb klasifikujeme jako šum, neboť neodpovídá charakteru pohybu srdeční stěny a způsobuje tak výkyvy ve spočteném signálu.

Segmentace pátého záznamu je zobrazena na obrázku 5.5. Kvalita tohoto záznamu je velice dobrá. Záznam obsahuje dobře rozeznatelnou srdeční stěnu i chlopeň. Šum je v tomto záznamu minimální. Signál má očekávaný periodický tvar s výraznými kladnými hroty. Všimněme si, že mezi těmito hroty má signál snahu narůstat. To je stejně jako v případě čtvrtého echokardiografického záznamu způsobeno výraznou chlopní. Rozsahy segmentů odpovídají očekávání, neboť jejich hranice jsou umístěny vždy na pozicích kladných hrotů signálu. Projekce hranic segmentů na EKG křivku nám vždy dává stejně jako v případě prvního a druhého echokardiografického záznamu ST úsek. ST úsek je fáze segmentace určena zatím u většiny záznamů a jedná se o úsek, ve kterém očekáváme umístění hranic segmentů. To je způsobeno tím, že srdce je v této fázi nejvíce stažené a tato vlastnost nám vzhledem k výpočtu signálu (kapitola 3) dává největší průměrnou vzdálenost.

Segmentace šestého záznamu spolu s EKG křivkou je zobrazena na obrázku 5.6. Kvalita záznamu je dobrá. Signál obsahuje výraznou srdeční stěnu i chlopeň a minimum šumu. Signál se v tomto případě podobá signálu ze čtvrtého echokardiografického záznamu, neboť i zde je velice výrazná mezikomorová chlopeň. S jejím vlivem na signál jsme se již seznámily u čtvrtého a pátého záznamu. Průběh signálu tak odpovídá periodicky se opakující dvojici hrotů. Segmentace opět přiřadila hranice podle dominantní frekvence, které odpovídá vždy



Obrázek 5.5: Signál a polohy hranic segmentů pátého echokardiografického záznamu.

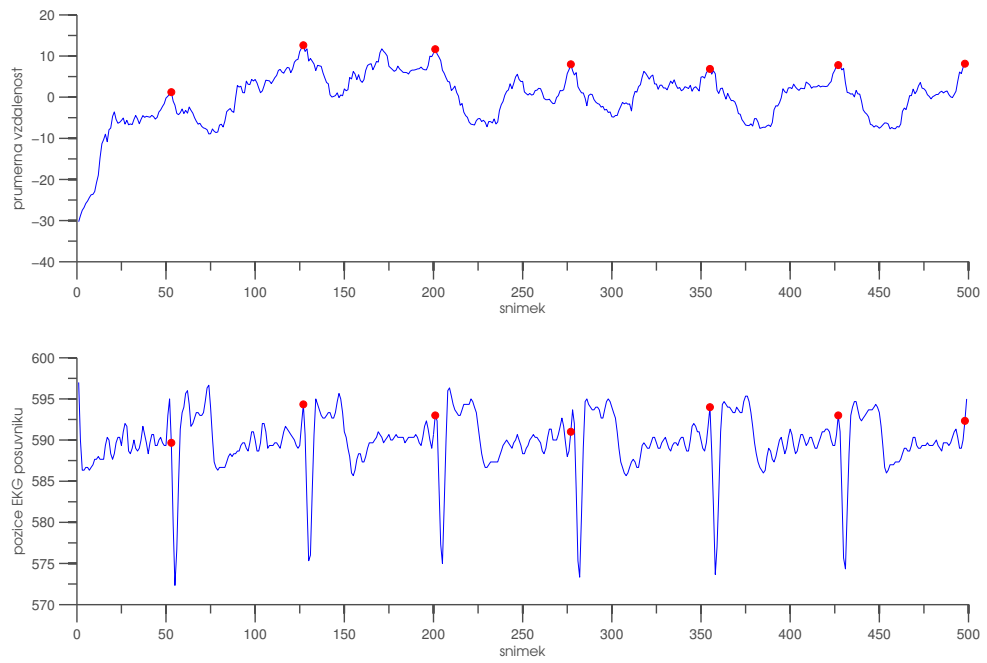
vyšší hrot. Na tomto snímku si lze ověřit, že pokud náš signál více odpovídá charakteru pohybu chlopně, tak srdeční fáze určené hranicemi segmentů budou odpovídat QRS komplexu. Stejně jako u čtvrtého snímku by správné řešení bylo umístit hranice segmentů na levý hrot signálu.

Signál ze sedmého echokardiografického záznamu spolu s rekonstruovanou EKG křivkou je zobrazen na obrázku 5.7. Kvalita tohoto záznamu je dobrá. Srdeční stěna je zde spolu s mezikomorovou chlopní výrazná. Sedmý testovaný záznam v sobě obsahoval mnoho šumu. V tomto případě průběh signálu odpovídá očekávanému periodicky se opakujícímu kladnému hrotu. I zde lze pozorovat vliv srdeční chlopně jako u předchozích záznamů. Jednotlivé segmenty leží správně v oblastech mezi hroty. Poloha jejich hranic na EKG křivce odpovídá úseku ST.

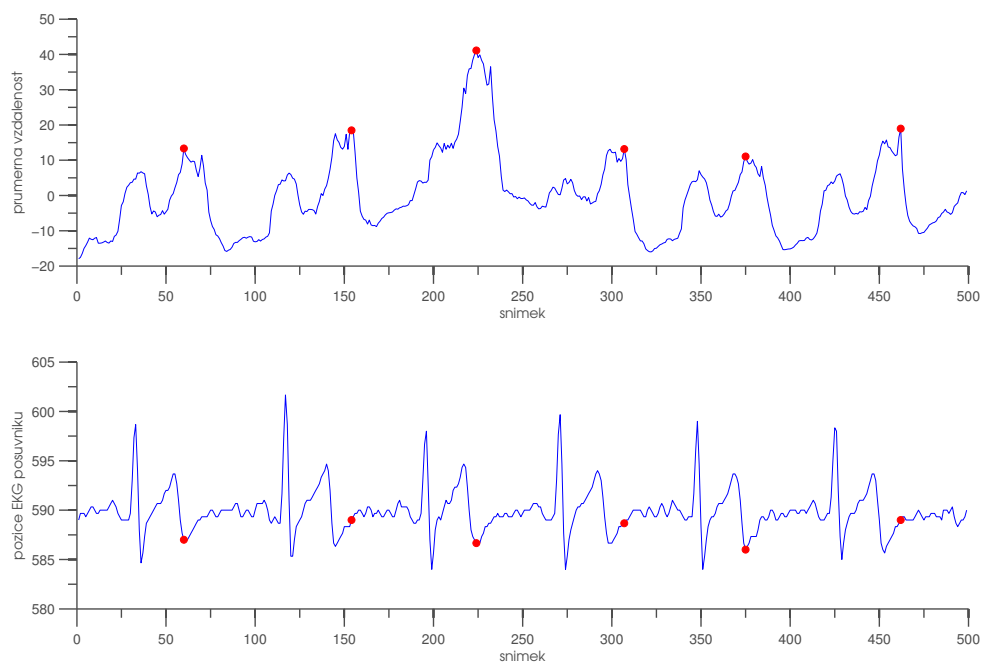
Vyhodnocení segmentace posledního echokardiografického záznamu je zobrazeno na obrázku 5.8. Kvalita tohoto záznamu je velice dobrá s velice výraznou srdeční stěnou i chlopní. Šum je v tomto záznamu minimální. Tvar signálu má tak očekávaný průběh. Segmenty jsou zde také správně určeny. Projekce polohy hranic segmentů na EKG křivku nám dává vždy úsek ST.

5.2 Poznatky získané z měření přesnosti

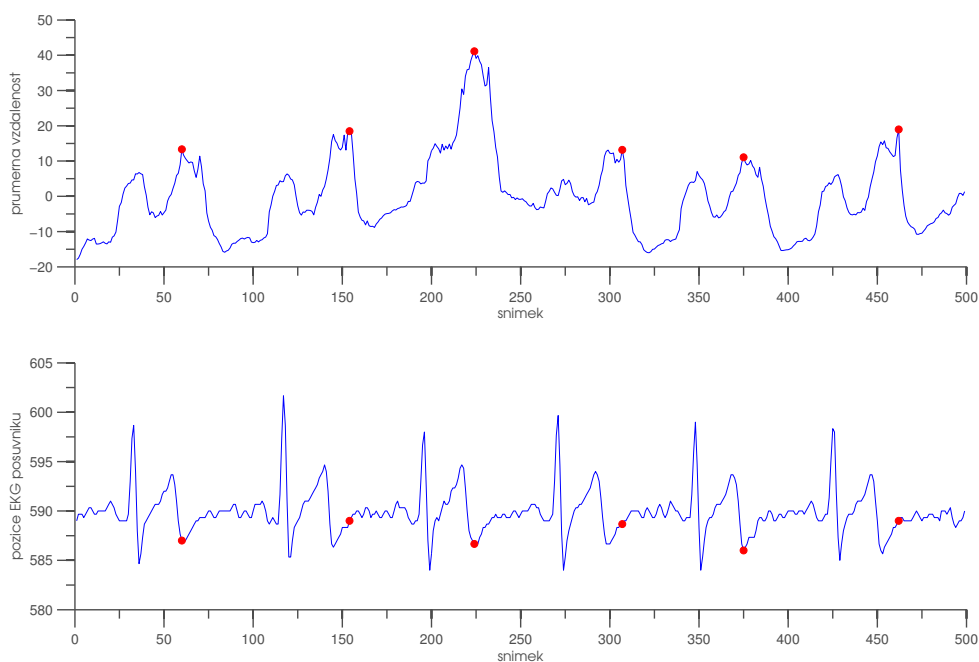
Výsledky z měření přesnosti segmentace jsou shrnuty v tabulce 5.1. Během měření byly získány následující poznatky: Segmentace je úspěšná pokud je srdeční stěna v echokardiografickém záznamu dobře viditelná. Segmentace proběhla na všech osmi záznamech s výjimkou jednoho, kde srdeční stěna byla nevýrazná. Dále zarovnání jednotlivých segmentů určené z pozice hranic segmentů na EKG křivce značně závisí na tom, jak výrazně se v echokardi-



Obrázek 5.6: Signál a polohy hranic segmentů šestého echokardiografického záznamu.



Obrázek 5.7: Signál a polohy hranic segmentů sedmého echokardiografického záznamu.



Obrázek 5.8: Signál a polohy hranic segmentů osmého echokardiografického záznamu.

ografickém záznamu pohybuje chlopeč. U záznamu, kde je pohyb chlopně značný, má náš algoritmus snahu sledovat více chlopeň než srdeční stěnu a tak i výsledný signál odpovídá pohybu chlopně. Běžně se tato anomálie projevuje vznikem dalších hrotů signálu. Hranice segmentů poté často leží v komplexu QRS na EKG křívce. Správné srdeční fázi, která by měla odpovídat hranicím segmentů, odpovídá fáze repolarizace komor ležící v úseku ST na EKG křívce.

	Srdeční stěna	Chlopeč	Segmentováno	Pozice na EKG
Záznam 1	Výrazná	Nevýrazná	Ano	ST
Záznam 2	Výrazná	Výrazná	Ano	ST
Záznam 3	Nevýrazná	Nevýrazná	Ne	-
Záznam 4	Výrazná	Výrazná	Ano	QRS
Záznam 5	Výrazná	Výrazná	Ano	ST
Záznam 6	Výrazná	Výrazná	Ano	QRS
Záznam 7	Výrazná	Výrazná	Ano	ST
Záznam 8	Výrazná	Výrazná	Ano	ST

Tabulka 5.1: Výsledky měření přesnosti segmentace echokardiografického záznamu.

Ke zlepšení námi navržené metody se prvně nabízí odstranění vlivu mezikomorové chlopně na charakteristiku pohybu srdeční stěny. Toto vylepšení může být provedeno několika způsoby. Prvním způsobem je využít rekonstruované EKG křívky a zjišťovat v jaké srdeční fázi dochází k segmentaci. Tento krok může být implementován algoritmem pro hledání segmentů (sekce 3.7). Algoritmus pro hledání segmentů bude nacházet QRS komplexy na EKG křívce, neboť je tento komplex největší výchytkou v elektrické aktivitě srdce. V případě, že

by hranice segmentů ležely uvnitř QRS komplexu, vybrala by se další výrazná frekvenční složka signálu charakteristiky pohybu srdeční stěny. Poté by se signál znovu segmentoval, ale již ne od maximální hodnoty signálu, nýbrž od nějakého maximálního bodu ležícího mezi dvěma QRS komplexy.

Sofistikovanější přístup k odstranění vlivu srdeční chlopně na charakteristiku pohybu srdeční stěny by mohl využívat tzv. Active Shape Models [7]. Tento přístup nám dovoluje vytvořit si šablonu tvaru srdeční stěny. Tato šablona se bude deformovat spolu se srdeční činností zachycenou v echokardiografickém záznamu. Deformace šablony je v Active Shape Models omezena a připouští pouze deformace, které se vyskytly při trénování modelu. Po natrénování tohoto modelu nám stačí rozmístit body sledované metodou Lucas-Kanade do této šablony a analyzovat jejich pohyb. Tímto se vyhneme vzniku šumu způsobeným nejen pohybem chlopní, ale i přítomností jiných artefaktů v echokardiografickém záznamu. Aplikací Active Shape Models na echokardiografické záznamy se zabývají například články [6] a [10].

Kapitola 6

Závěr

Podařilo se mi navrhnout a implementovat algoritmus pro automatickou segmentaci echokardiografického záznamu podle srdečních stahů. Tento algoritmus je založen na metodě Lucas-Kanade, jejímž prostřednictvím je získána charakteristika pohybu srdeční stěny v průběhu záznamu. Tuto charakteristiku poté analyzuji pomocí algoritmu rychlé Fourierovy transformace, která nám tento signál převede do frekvenční domény. Ve frekvenční doméně určím dominantní frekvenci, jež odpovídá střídání srdečních stahů a poté segmentuji daný echokardiografický záznam podle periody této frekvence.

Pro testování algoritmu jsem využil možnosti rekonstrukce EKG křivky z dostupných echokardiografických záznamů. Testování odhalilo zejména citlivost algoritmu na pohyb mezikomorové chlopně, která způsobuje ve vypočtené charakteristice šum. Odstranění tohoto šumu bylo diskutováno v sekci 5.2, ve které nabízím dvě možnosti jak tento problém řešit (Active Shape Models a dvouprůchodová segmentace). Úspěšnost segmentace dostupných záznamů je dobrá. Pouze záznamy s velmi výraznou chlopní a velmi nevýraznou srdeční stěnou se nepodařilo správně segmentovat.

Z testů tohoto algoritmu plyne, že segmentace založená na optickém toku a diskrétní Fourierově transformaci je dostupnou variantou pro analýzu pohybu v echokardiografických snímcích. Pro využití v nemocničním prostředí, kde by tato metoda ušetřila práci specialisty, je vhodné dopracovat vylepšení diskutovaná výše. Dále se nabízí možnost změnit rovnici 3.26 tak, aby lépe zachycovala pohyb srdeční stěny. Jiná rovnice by například mohla namísto vzdálenosti počítat úhel daný orientací nějakého průměrného vektoru. Tento průměrný vektor bude spočítán z optického toku a jeho úhel se bude měnit podle aktuální srdeční fáze, ve které se echokardiografický záznam nachází.

Literatura

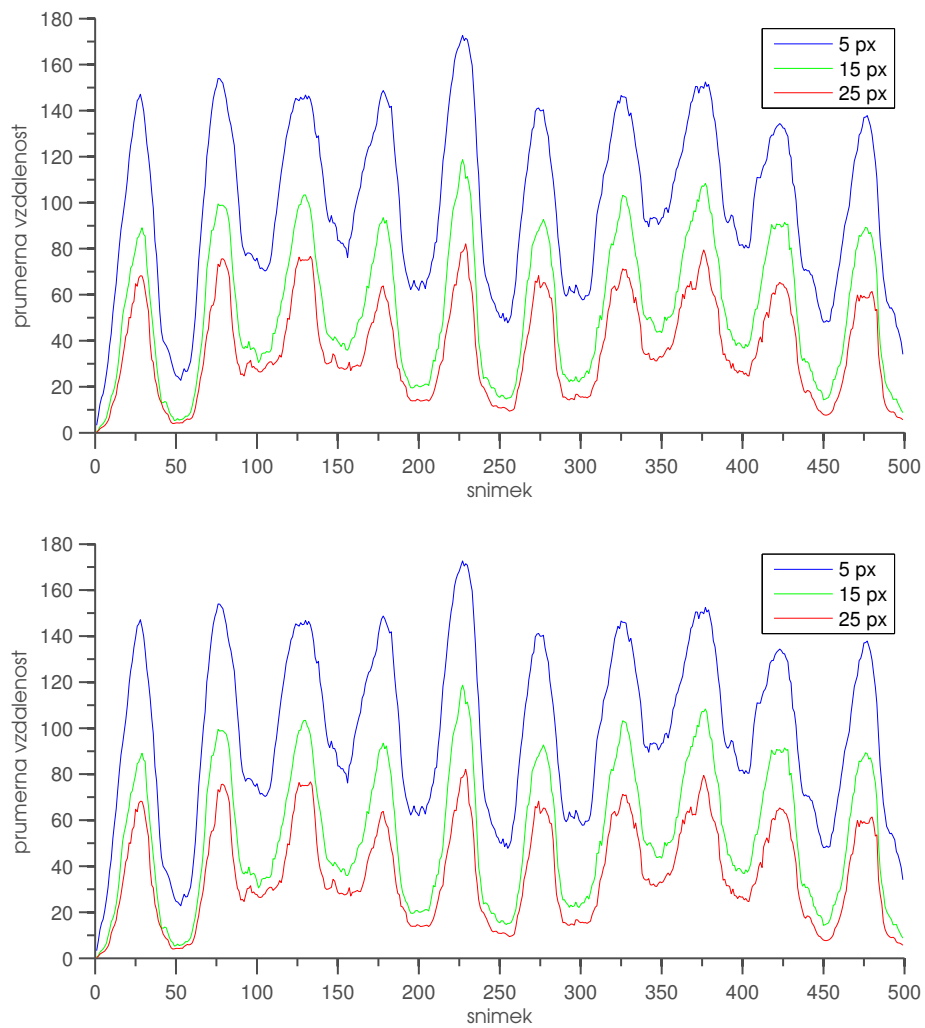
- [1] Adelson, E. H.; Anderson, C. H.; Bergen, J. R.; aj.: Pyramid methods in image processing. *RCA engineer*, ročník 29, č. 6, 1984: s. 33–41.
- [2] Bouguet, J.-Y.: Pyramidal Implementation of the Lucas Kanade Feature Tracker. *Intel Corporation*, 2001.
- [3] Bradski, G.; Kaehler, A.: *Learning OpenCV: Computer vision with the OpenCV library*. "O'Reilly Media, Inc.", 2008.
- [4] Chen, W.; Shi, Y. Q.; Xuan, G.: Identifying computer graphics using HSV color model and statistical moments of characteristic functions. In *Multimedia and Expo, 2007 IEEE International Conference on*, IEEE, 2007, s. 1123–1126.
- [5] Cooley, J. W.; Tukey, J. W.: An algorithm for the machine calculation of complex Fourier series. *Mathematics of computation*, ročník 19, č. 90, 1965: s. 297–301.
- [6] Cootes, T. F.; Hill, A.; Taylor, C. J.; aj.: The use of active shape models for locating structures in medical images. In *Information Processing in Medical Imaging*, Springer, 1993, s. 33–47.
- [7] Cootes, T. F.; Taylor, C. J.; Cooper, D. H.; aj.: Active shape models-their training and application. *Computer vision and image understanding*, ročník 61, č. 1, 1995: s. 38–59.
- [8] Frigo, M.; Johnson, S. G.: FFTW user's manual. *Massachusetts Institute of Technology*, 1999.
- [9] Harris, C.; Stephens, M.: A combined corner and edge detector. In *Alvey vision conference*, ročník 15, Manchester, UK, 1988, str. 50.
- [10] Hill, A.; Taylor, C. J.: Model-based image interpretation using genetic algorithms. In *BMVC91*, Springer, 1991, s. 266–274.
- [11] Hlaváč, V.; Sedláček, M.; vysoké učení technické v Praze, Č.: *Zpracování signálů a obrazů*. České vysoké učení technické, 2000.
- [12] Horn, B. K.; Schunck, B. G.: Determining optical flow. In *1981 Technical Symposium East*, International Society for Optics and Photonics, 1981, s. 319–331.
- [13] Li, Y.: Reforming the theory of invariant moments for pattern recognition. *Pattern Recognition*, ročník 25, č. 7, 1992: s. 723–730.

- [14] Lucas, B. D.; Kanade, T.; aj.: An iterative image registration technique with an application to stereo vision. In *IJCAI*, ročník 81, 1981, s. 674–679.
- [15] Mourek, J.: *Fyziologie*. Grada Publishing a.s., 2012.
- [16] Mysliveček, J.: *Základní fyziologické principy*. Vydavatelství ČVUT, 2002.
- [17] Nevřiva, P.: *Analýza signálů a soustav*. BEN, 2000.
- [18] Ondráček, O.: *Signály a systavy*. Slovenská technická univerzita, 1999.
- [19] Silbernagl, S.; Lang, F.: *Atlas patofyziologie člověka*. Grada Publishing a.s., 2001.
- [20] Trojan S. a kolektiv: *Lékařská fyziologie*. Grada Publishing a.s., 2003.

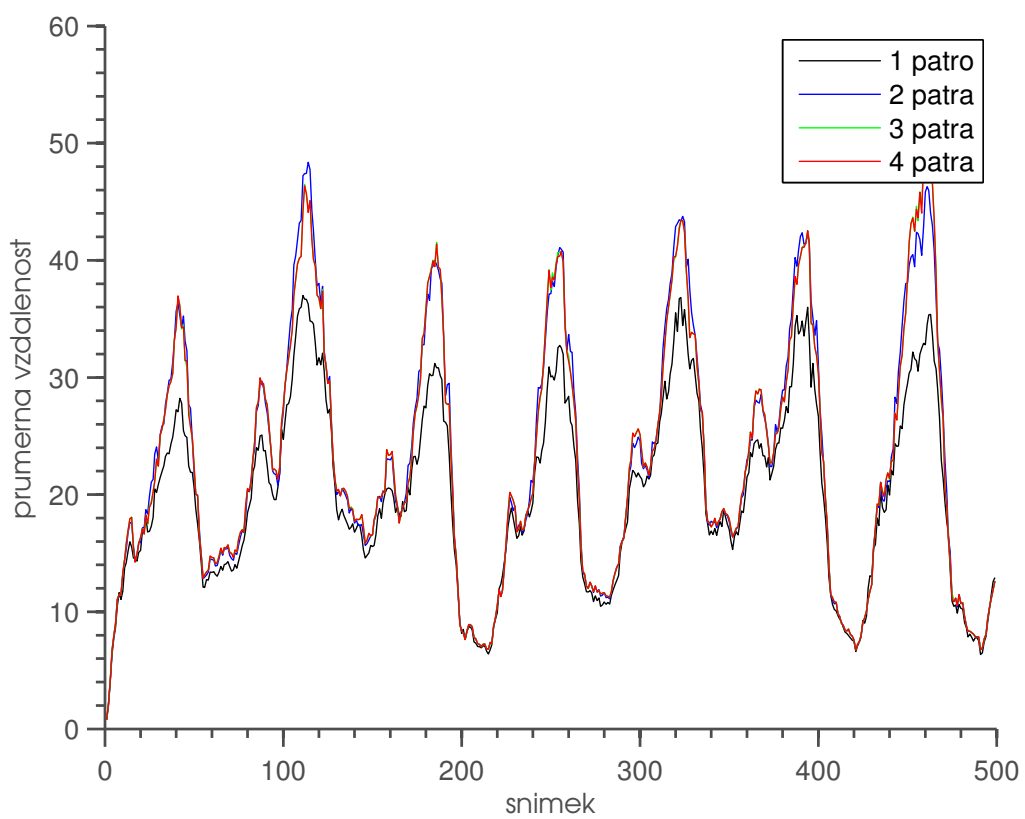
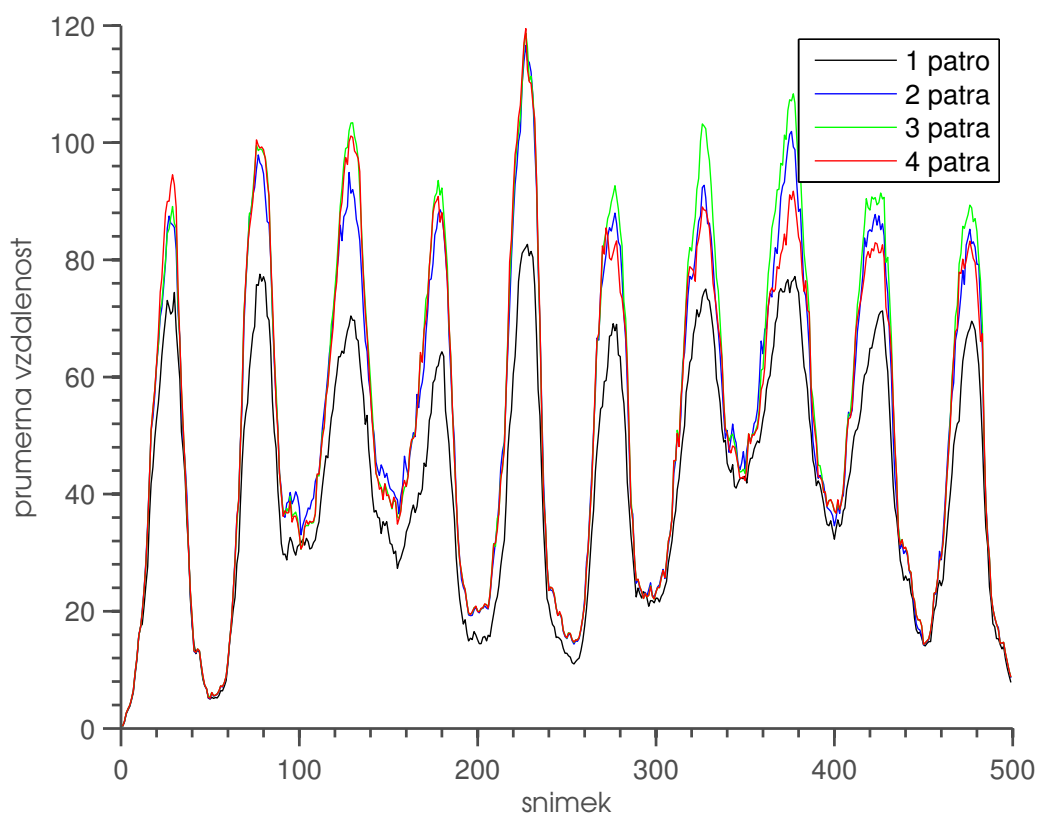
Příloha A

Test parametrů optického toku

Zde přikládám výsledky zbylých kalibrací parametrů funkce `calcOpticalFlowPyrLK()`. Tyto testy se týkají nastavení velikosti okna optického toku a počtu úrovní pyramidy snímků. Metodologie testování a způsob hodnocení testů je popsán v sekci 4.2.



Obrázek A.1: Kalibrace podle velikosti okénka optického toku.



Obrázek A.2: Kalibrace podle počtu úrovní pyramid snímků.

Příloha B

Obsah DVD

- Data - Složka Data obsahuje devět echokardiografických záznamů, které byly využity pro testování námi navrženého algoritmu. Část šestého záznamu je vadná a tedy tento záznam nebyl zahrnut do testování.
- HeartBeat - Složka HeartBeat obsahuje C++ program jako projekt ve Visual Studiu. Tento program implementuje zadání této práce.
- MatlabScripts - Složka MatlabScripts obsahuje pomocné skripty v Matlabu, kterých jsem využil při řešení této práce. Je zde umístěn např. validační skript `eval.m`, který jsem použil při testování programu HeartBeat. Další zajímavý skript je například `segmentaceVID.m`, který zobrazuje průběh signálu a jednotlivé segmenty zároveň se záznamem z příslušného vyšetření.
- TestParam - Složka TestParam obsahuje program v jazyce C++ v prostředí Visual Studio. Tento program testuje parametry optického toku nad dostupnými echokardiografickými záznamy.
- Tex - Složka Tex obsahuje tento text v prostředí \LaTeX .
- TrackEKG - Složka TrackEKG obsahuje C++ program v prostředí Visual Studio. Tento program rekonstruuje EKG křivku ze zvoleného echokardiografického záznamu.