



Technická zpráva projektu č. VI20172020068

**Nástroje a metody zpracování videa a obrazu pro zvýšení
efektivity operací bezpečnostních a záchranných složek
(VRASSEO)**

Zařízení pro snímání zbraní

Tomáš Goldmann, Michal Dvořák, Martin Drahanský

Fakulta informačních technologií
Vysokého učení technického v Brně
Božetěchova 1/2
612 66 Brno, Česko

Prosinec 2017

Zařízení pro snímání zbraní

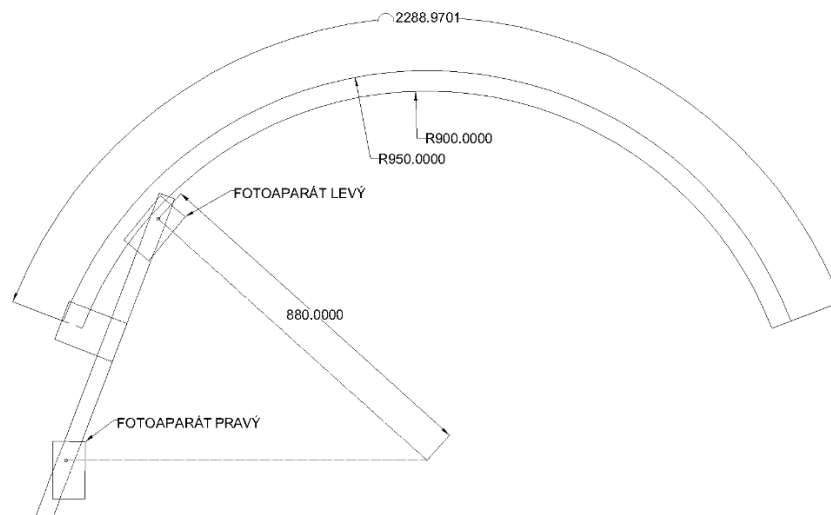
Autor: Tomáš Goldmann, Michal Dvořák, Martin Drahanský

Pro účely vývoje algoritmů pro detekci zbraní, je třeba vytvořit databázi zbraní, která bude využita pro metody strojového učení. Vzhledem k plánovanému nasazení této detekce musí být databáze tvořena s ohledem na budoucí orientační invarianci – tedy v reálných podmínkách a různorodém přirozeném prostředí, ve všech představitelných orientacích a pozicích. Databáze bude tvořena daty pro vytvoření 3D modelů zbraní. Tímto způsobem získáváme absolutní kontrolu nad kvalitativní úrovní databáze, a to díky možnosti kombinovat pozadí, orientaci a pozici zbraně, čímž jsme schopni vytvořit databázi takřka libovolného rozsahu, což by při manuálním sběru bylo neproveditelné.

Princip vytváření 3D modelů vychází z multimodální datové syntézy, kde kombinací hloubkových a prostých optických snímačů shromažďujeme množinu informací, jako například texturní informace o snímaném objektu a hloubkovou mapu s objektem ve scéně, které využijeme při rekonstrukci věrného modelu, kde jsou jednotlivé metody 3D analýzy vzájemně syntetizovány za účelem zvýšení kvality snímání. V tomto případě specificky spojujeme texturní informace, hloubková data z hloubkové kamery, a hloubková data dopočítaná z metod stereo-vidění.

1. Konstrukce zařízení

Zařízení pro snímání zbraní (dále „U-rampa“) je hliníkový čtvercový profil tvarovaný do půlkruhu, po kterém se pohybuje vozík se snímacími senzory. Aktuální verze je tvořena obloukem kružnice, který je menší než oblouk poloviny kruhu. Ilustrační náčrt zařízení je na snímku níže.



Obrázek 1 Náčrt koncepce U-rampy.

Realizace pohybového modulu

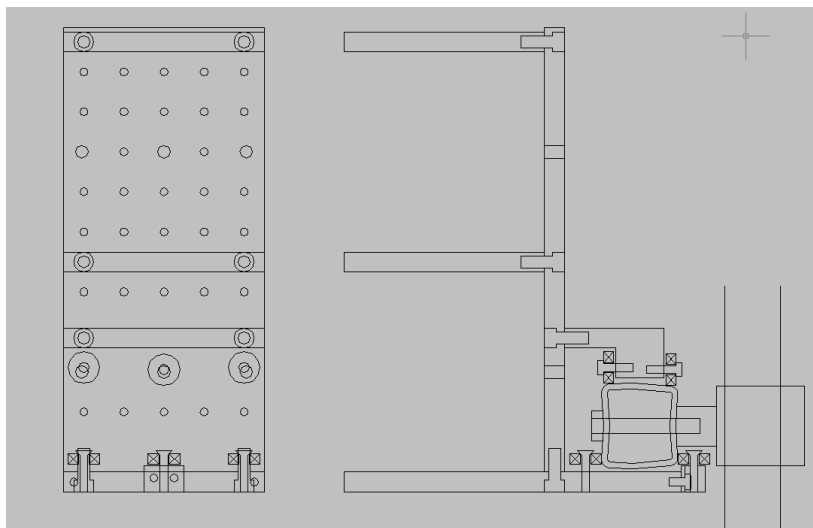
Jak již bylo uvedeno, jeden zdroj hloubkových dat vychází z principu stereo-vize, kde napříč množinou obrazových dat hledáme významné body, ke kterým přiřazujeme řešením korespondenčního problému odpovídající párový bod v obraze jiném, a následně z měřeného rozdílu v obraze dopočítáme informaci hloubky. Aby tato metoda byla aplikovatelná, bylo třeba vytvořit generátor této obrazové množiny, vzhledem k požadavku na co nejvyšší detail eventuálního modelu. Tato množina musí být pro každou zbraň dostatečně velká, neboť manuální přístup by byl časově náročný, a použitelnost takovýchto dat kvůli nekalibrované pozici snímacího zařízení nejistá. Pro účely projektu bylo zkonstruováno polohovací zařízení, které obstarává poloautomaticky snímání jednotlivých zbraní, a to polohováním snímacích zařízení.

Jak bylo uvedeno výše, trasa pohybu je vedena za využití hliníkového profilu (viz Obrázek 1) zaujímavějšího tvar výseče kružnice. Na tomto profilu se nalézá vozík, na který jsou umístěna snímací zařízení, a zároveň obsahuje pohonnou řídicí a výpočetní jednotku, díky které realizuje pohyb po hliníkovém rameně.

Vozík

Obrázek 2 znázorňuje náčrtek, kde si můžeme udělat lepší představu o principu fungování posuvného vozíku. V pravé části vidíme náčrtek z profilu, reprezentující uchycení vozíku k hliníkovému profilu U-rampy. Pohonná jednotka je pak uchycena zespodu tohoto vozíku, na rotoru připevněné pogumované kolečko, které se nachází v kontaktu s profilem, a jejím otáčením je docíleno pohybu vozíku.

Na vozíku si můžeme všimnout precizních horizontálních plošin, které slouží k umístění řídicí jednotky pohonného systému a snímacích komponent. Jmenovitě, řídicí jednotka včetně výkonové části pohonného systému je umístěna na spodní horizontální plošině, zatímco k prostřední horizontální plošině je připojeno sensorické rameno, ke kterému jsou přimontována jednotlivá sensorická zařízení (viz Kapitola 2). Rameno bylo připraveno tak, aby se jednotlivá zařízení nacházela vůči vozíku v pevné kalibrované poloze, díky čemuž získáváme apriorní informace použitelné pro účely následující rekonstrukce obrazu.



Obrázek 2 Vozík U-rampy.

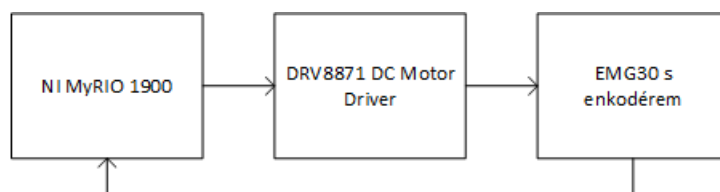
Pohonná jednotka

Jak bylo zmíněno v části o vozíku, jeho lineární pohyb je realizován převodem rotačního pohybu pohybové jednotky přes pogumované kolečko po hliníkovém profilu. Vzhledem k váze vozíku a snímacím přístrojům bylo třeba vybrat dostatečně naddimenzovaný pohon se zpětnovazebními

možnostmi. Pro tyto účely byl zvolen motor EMG30 s převodem od společnosti Robot Electronics. Tento motor se jmenovitým točivým momentem $1,5 \text{ kg}\cdot\text{cm}^{-1}$ nám umožňuje díky poloměru pogumované kolečka 0,5 cm bezpečně sunout až 3 kg nákladu, což je pro naše účely dostačující.

Druhý důvod pro volbu EMG30 je integrovaný enkodér. Díky tomuto výstupu jsme schopni do našeho polohovacího algoritmu implementovat zpětnovazební řízení, díky němuž jsme schopni docílit výše zmíněného kalibrovaného sběru dat. Jelikož se jedná o výkonovou pohonnou jednotku, bylo třeba obstarat vhodný ovladač, který bude schopen uspokojit maximální odběr 2,5 A při rozjezdu. Pro tento účel byl zvolen DRV8871 DC Motor Driver, schopný uspokojit až 3,6 A nárazově a 2.0 A dlouhodobě, což představuje bezpečný ovladač pro náš účel.

Poslední částí řetězce je řídicí jednotka, která je schopna reagovat na výstup z enkodéru a realizovat řídicí algoritmus. Obrázek 3 znázorňuje princip fungování. Realizaci zpětnovazebního řízení se budeme věnovat ve vlastní sekci.



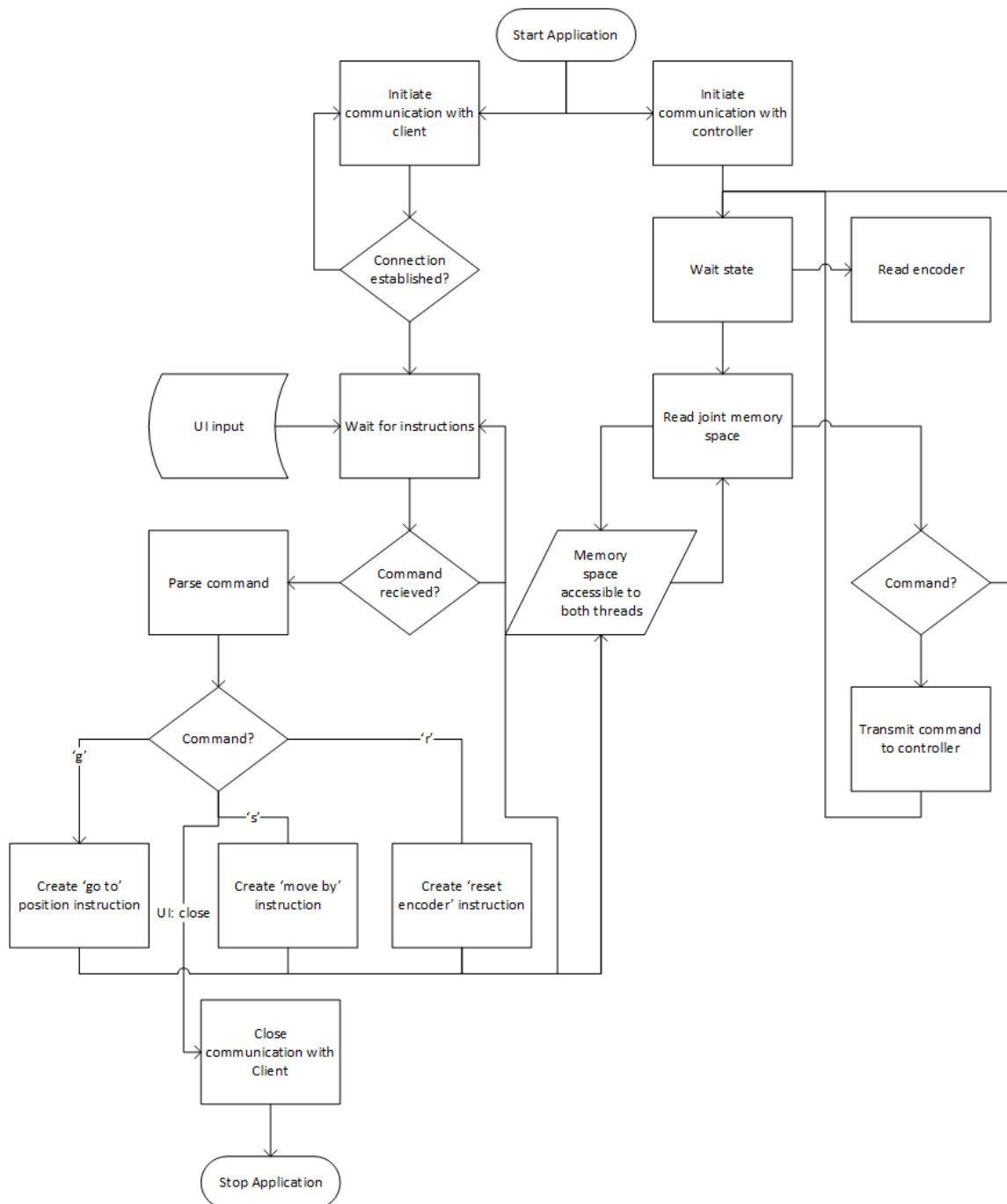
Obrázek 3 Fungování pohybového modulu.

Řízení vozíku U-rampy

Pohyb vozíku je důležitou částí celého systému, jak bylo objasněno výše. V následující části tak pokračuje popis jednotlivých částí softwarového a hardwarového řešení. Pro přehlednost rozdělíme systém na embedded část, která přijímá příkazy z nadřazeného systému na PC, a PC část která realizuje komunikaci mezi klientem, PC a řídicí platformou.

Komunikační PC platforma

Pro účely snímání se v blízkosti U-rampy musí nacházet dedikované PC. Na tomto PC se zároveň nachází komunikační rozhraní, skrze které jsme schopni komunikovat s řídicí platformou. V tomto programu jsou paralelně spuštěny dvě smyčky. První smyčka obsluhuje komunikaci s případným nadřazeným klientem. Druhá smyčka pak na základě vstupů z uživatelského prostředí či nadřazeného systému instruuje embedded část. Obrázek 4 obsahuje princip fungování komunikační platformy.



Obrázek 4 Stavový diagram komunikačního rozhraní.

Jak je z diagramu zřejmé, jeho fungování je podmíněno připojením nadřazeného klienta, se kterým komunikuje pomocí protokolu TCP/IP. Nadřazený klient pomocí implementovaného protokolu a jednoduchých příkazů posílá instrukce, které jsou pak v rámci druhé smyčky odesílány embedded zařízením. Vzhledem k využití dvouvláknového paralelismu musela být realizována komunikace mezi vlákny.

Komunikační protokol využívá jednoduchého formátu, kde je **7 bytů** odesíláno v následujícím pevném formátu.

| | | | | | | |
|---|---|---|---|---|---|---|
| I | D | X | X | X | X | X |
|---|---|---|---|---|---|---|

- I představuje instrukci, která může nabývat veličin „g“ – běž na pozici x, „s“ – realizuj posun o x, „r“ – restartuj systém nebo „i“ – inicializuj systém.
- D indikuje směr hodnotami „1“ – doleva či „0“ – doprava.
- A parametry „x“ jsou parsovány jako numerické hodnoty pro instrukce „g“ a „s“.

Pro univerzálnost tohoto systému a ulehčení vývoje není nadřazený systém nijak limitován a může jej nahradit jiný bez omezení.

Firmware řídicí platformy

V reakci na instrukce odeslané z komunikační platformy se mění stav řídicí jednotky. Po interpretaci přijaté instrukce může nastat jeden ze tří stavů:

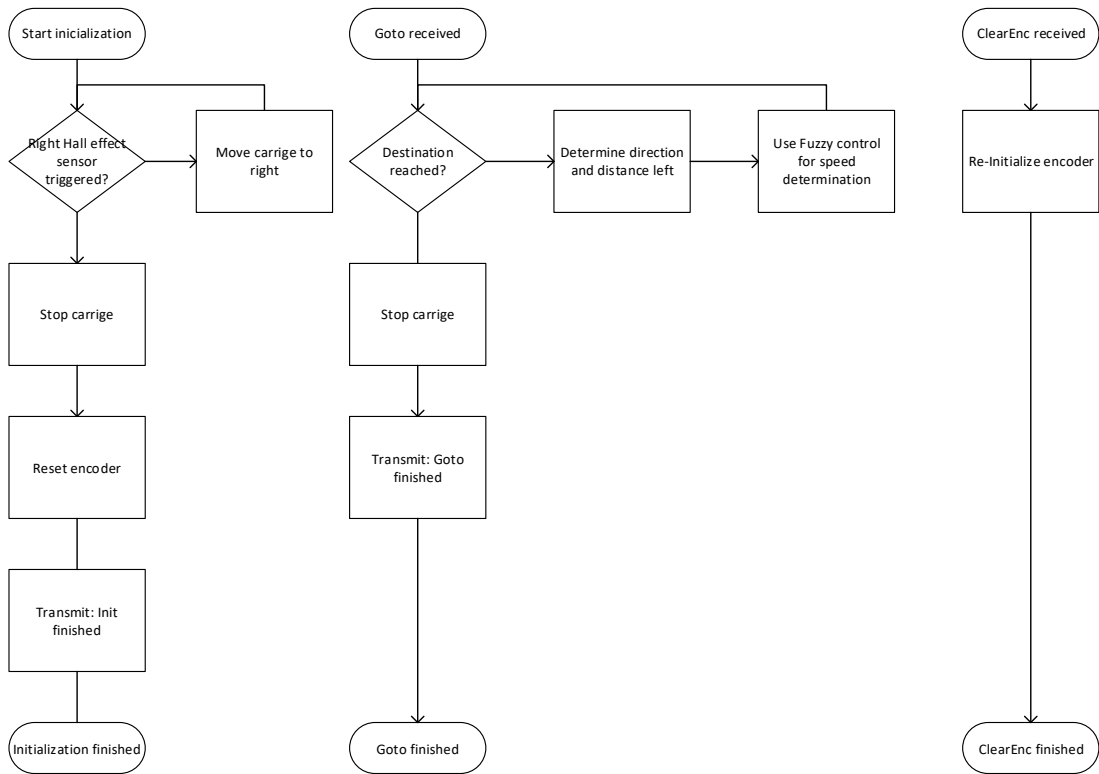
1. Řídicí platforma bude inicializována. V tomto případě zajede vozík na krajní pozici rampy, kde zastaví a vynuluje enkodér.
2. Řídicí platforma změní svoji pozici pojezdem na zadanou polohu či o zadanou vzdálenost.
3. Řídicí platforma vyresetuje enkodér.

Princip fungování těchto stavů je vyobrazen na Obrázku 5. V průběhu vykonávání těchto funkcí je systém uzamčen a do jejich dokončení nepřijímá další instrukce. Po jejich dokončení, ať již regulérním či předčasným, je komunikační platformě odeslána zpráva o dokončení operace, která tuto instrukci přepoše případnému nadřazenému systému.

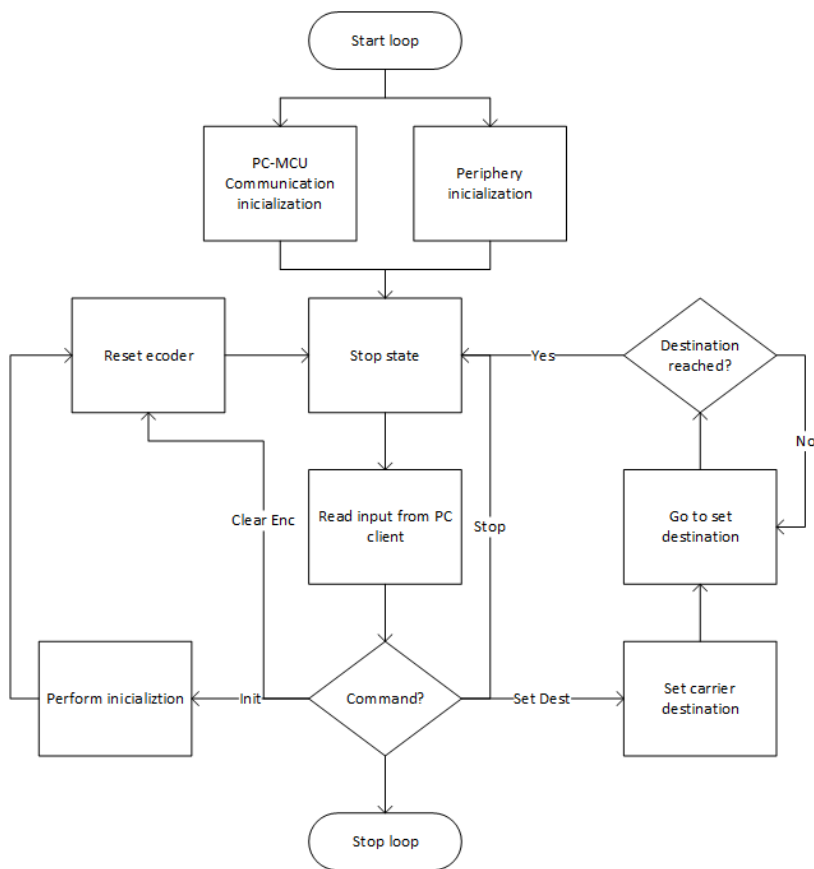
Funkcionalita GOTO (Změň pozici) obsahuje důležitou funkcionalitu, a to fuzzy regulaci. Fuzzy regulace je jedním ze způsobů řízení. Tato metoda se odlišuje od mnoha dalších svým až stochastickým přístupem, kde vstup (v našem případě vzdálenost zbývající k zadané pozici), je takzvaně fuzzyfikován, což představuje proměnu na váženou symbolickou veličinu. Tato proměnná je následně přepočítána na symbolickou výstupní veličinu dle definované převodní funkce, která je poté defuzzyfikována do výstupní hodnoty, která v našem případě vyjadřuje rychlost motoru.

Tento způsob regulace byl zvolen s ohledem na časté změny jednotlivých snímacích jednotek a dalších vlastností zařízení. Zatímco klasické řídicí metody, jako PID regulace, by po každé změně vyžadovaly přepočítání jednotlivých parametrů, fuzzy regulaci vycházející ze symbolických veličin jako „Daleko“ a „Rychle“, lze jednoduše modifikovat na základě aktuálního stavu.

Stavový diagram reprezentující obecné chování a přecházení mezi jednotlivými stavy lze pozorovat na Obrázku 6.



Obrázek 5 Funkcionalita řídicí platformy.



Obrázek 6 Stavový diagram embedded zařízení.

2. Zařízení pro snímání dat

Cílem snímání dat pro vytváření 3D modelů zbraní je získat kvalitní data, která umožní další strojové zpracování. Data by vzhledem k povaze měla co nejlépe zachytit potřebné detaily zbraně tak, aby bylo možné určit charakteristické rysy daného typu zbraně. Tato data mohou být užitečná pro pokročilé detektory zbraní, které dokáží určit i o jaký typ zbraně se jedná.

Při experimentech jsme vyzkoušeli několik přístupů k získání 3D modelů. Prvních z nich bylo využití hloubkových snímačů založených na promítání vzoru prostřednictvím IR senzoru. Nejznámějším senzorem, který tento přístup využívá, je zařízení od firmy *Microsoft – Kinect*. Na trhu jsou aktuálně dostupné dvě verze; první může zachytit hloubkovou mapu v rozlišení 640×480 pixelů, zatímco druhá verze pracuje s rozlišením 512×424 pixelů. Zařízení Kinect dále obsahuje RGB kameru, která má senzor s rozlišením 640×480, resp. 1.920×1.080 pixelů [2]. Pro rekonstrukci jsme vyzkoušeli několik programů, které na základě hloubkových a obrazových dat prováděly vytváření 3D modelů. Lesklý povrch zbraně bohužel snižoval kvalitu hloubkových dat, což zapříčinilo to, že 3D modely byly nekvalitní a poměrně generické.

Mezi další testované senzory patří kamera *Intel RealSense F200*, *DepthSense DS325* a *Sense 3D Scanner*. Provedením několika experimentů jsme zjistili, že pro vytváření 3D modelů zbraní není tento přístup příliš vhodný. Problém nastával při navazování jednotlivých hloubkových dat na sebe, při některých úhlech dokonce došlo k „rozsypaní“, a tím pádem k nevygenerování 3D modelu. Zlepšení výstupu přineslo nanesení zmatňujícího spreje, který snížil odrazivost kovového povrchu zbraně.

Na základě poznatků z experimentů byl navržen pojízdný vozík se třemi senzory pro pořízení dat, z kterých lze vytvořit kvalitní databázi zbraní.

Snímání 2D snímku

Nejllepších 3D modelů zbraně jsme dostali rekonstrukcí z 2D fotek. U-rampa nám umožňuje provádět pozicované pořizování obrazových dat vůči objektu (v našem případě zbraní). Díky půlkruhové konstrukci zařízení můžeme u každého snímku určit z jakého úhlu byl pořízen.

Ke snímání používáme 2 fotoaparáty *Canon EOS 80D*, které jsou umístěny na rameni ve vzdálenosti 63 cm mezi stativovými šrouby. Podle konstrukce (viz Obrázek 1) je U-rampa tvořena obloukem půlkruhu, tudíž je vzdálenost mezi fotoaparáty vypočítaná tak, aby pokryla půlkruh, byť na menší kružnici. Pro testování byl zvolen objektiv *Canon EF-S 18-55mm f/4-5.6*, který je pro účely tohoto zařízení dostatečný. Objektiv je v režimu manuálního ostření, protože při automatickém ostření se stávalo, že došlo k zaostření pozadí místo zbraně. Natočení fotoaparátu je určeno na základě poloměru U-rampy tak, aby souběh optických os obou fotoaparátů byl v potenciálním středu zbraně. Dvojice fotek pak vytváří stereoskopické snímky, ze kterých je možné vypočítat mapu disparity. Jelikož se snímání provádí po kruhu, je možné disparitní mapy počítat z různých dvojic snímků.

Pro experimentální rekonstrukci 3D modelu používáme software *3D Zephyr Lite*, který poskytuje kvalitní 3D modely z 2D fotek. V porovnání s 3D skenery se jedná o podstatně lepší výsledky.

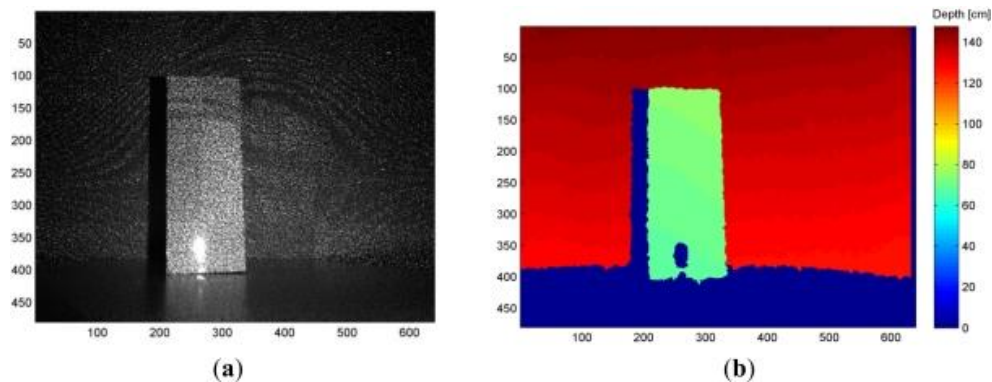
Zařízení pro pořizování video záznamu

U-rampa využívá k pořízení videozáznamu z průjezdu vozíku RGB senzor na zařízení *Microsoft Xbox One Kinect*, které dokáže pořídit záznam v rozlišení Full HD (až 30 FPS za dobrých světelných podmínek). Videozáznam slouží k doplnění dat z fotoaparátu. Při reálném zpracování záznamu závisí snímkovací frekvence na rychlosti enkódování jednotlivých snímků. Proto je zapotřebí výkonný PC, který dokáže zpracovávat data ze všech senzorů.

Senzor pro zisk hloubkový dat

Hloubková data získáváme z hloubkového senzoru zařízení Kinect. Jelikož jsme po prvotních experimentech s využitím hloubkových dat nedosahovali uspokojivých 3D modelů, rozhodli jsme se tento typ dat přidat jako další doplněk ke 2D snímkům.

Model Kinect verze 1 pracuje na bázi promítání strukturovaného světla do prostoru, které je následně snímáno IR snímačem. Na základě korelace původního promítaného vzoru s nasnímaným vzorem se vytváří hloubková mapa [1]. Když je vzor promítán na objekt, jehož vzdálenost je menší či větší od referenční pozice vzoru, bude výsledný vzor v infrasnímku posunutý vůči optické ose projektoru a centrem perspektivy IR kamery.



Obrázek 7 Znárodnění vzorů promítaného IR projektorem a následná hloubková mapa [1].

V zařízení *Microsoft Xbox One Kinect*, které používáme ve výsledné verzi snímacího zařízení, je podobný typ senzoru, který je ovšem založený na technologii time-of-flight (ToF). U jednotlivých bodů se provádí určení vzdálenosti na základě doby lety fotonu a jeho návratu do IR snímače RTT (round-trip time). Technické vlastnosti senzoru: $512 \times 424,30$ Hz, FOV: 70×60 , One mode: 0,5–4,5 metrů [3].

3. Software pro snímání dat

Důležitou součástí zařízení je softwarová výbava, která je tvořena dvěma aplikacemi. Prvních z nich slouží pro řízení vozíku (viz Komunikační PC platforma) a druhá pro zpracování dat ze senzoru. V této části si rozebereme software pro zpracování dat. Aplikace je napsaná v programovacím jazyce C# a zajišťuje veškeré zpracování dat. Aplikace komunikuje prostřednictvím TCP/IP s aplikací pro řízení posunu vozíku (viz Řízení vozíku U-rampy). Výstupem aplikace jsou snímky z jednotlivých úhlů, XML informace k danému typu zbraně, PCL (*Point Cloud Library*) data pro jednotlivé úhly a video z průjezdu U-rampy v rozlišení Full HD.

Komunikace s fotoaparáty Canon EOS

Fotoaparáty pro pořizování snímku byly vybrány s ohledem na dostupnost SDK (v případě Canon EOS) pro programovací jazyk C. Komunikace mezi fotoaparáty a PC probíhá prostřednictvím rozhraní USB 2.0, které umožňuje obousměrnou komunikaci mezi oběma zařízeními. Ačkoliv existuje SDK DLL knihovna pro jazyk C, je na internetu dostupná i neoficiální *wrap* pro programovací jazyk C# s názvem Canon EDSDK [4]. Tato knihovna obsahuje především funkce pro:

- Živý náhled (Liveview)
- Pořízení snímku
- Získání/nastavení ISO, Av, Tv

- Nahrávání videa
- Stahování filmů/fotek do PC

Tyto vlastnosti jsou pro nás klíčové z hlediska dálkového řízení fotoaparátu přes PC. Knihovna je strukturovaná do několika tříd:

- **CanonAPI:** Hlavní třída, která má na starosti životní cyklus SDK, připojené kamery a události.
- **Camera:** Třída pro komunikaci s fotoaparátem. Obsahuje metody pro nastavení vlastností, pořízení fotky, stáhnutí fotky do PC a další.
- **CanonSDK:** Třída obsahující volání na nativní Canon SDK DLL.
- **STAThread:** Pomocná třída k vytvoření STA vlákna.
- **ErrorHandler:** Třída se statickými metodami pro zpracování chyb.

Snímání probíhá tak, že fotoaparát na základě předešlého nastavení (ISO, Av, Tv) provede pořízení snímku, který následně uloží na SD kartu. Pokud je prostřednictvím SDK nastaven režim stahování do PC, je po vyfocení vyvolána událost, prostřednictvím které je vrácen snímek z fotoaparátu pro další zpracování.

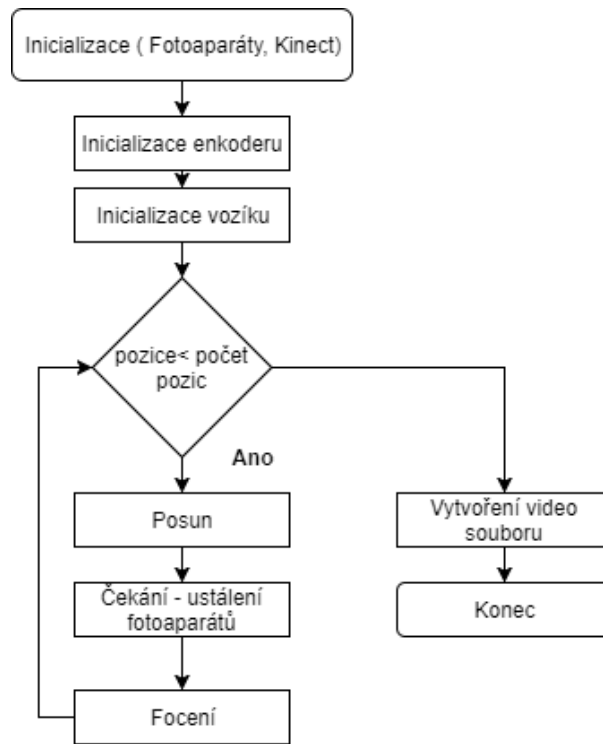
Komunikace se zařízením Kinect

Firma Microsoft poskytuje SDK pro komunikaci s *Microsoft Xbox One Kinect* (dále Kinect) v programovacím jazyce C#. SDK nám umožňuje získávat hloubková data a obrazová data ze zařízení Kinect. Pokud Kinect zpracuje snímek, ať už hloubkový či s obrazovými daty, dojde k vyvolání události připojené přes `MultiSourceFrameArrived` nebo `FrameArrived` (třída `ColorFrameReader`) a `FrameArrived` (třída `DepthFrameReader`). V události `MultiSourceFrameArrived` máme k dispozici hloubkový i obrazový snímek, který můžeme dále zpracovávat. Událost `FrameArrived` (pro třídu `DepthFrameReader` a `ColorFrameReader`) pak obsluhuje hloubková data a obrazová data separátně. Z hlediska zpracování dat je klíčová transformace do požadovaných formátů. V případě hloubkových dat se provádí převod do PCL souboru, zatímco v případě obrazových dat se provádí převod na *Bitmap*, která se dále zpracovává v enkodéru pro vytvoření videosouboru.

Implementace a princip funkce aplikace

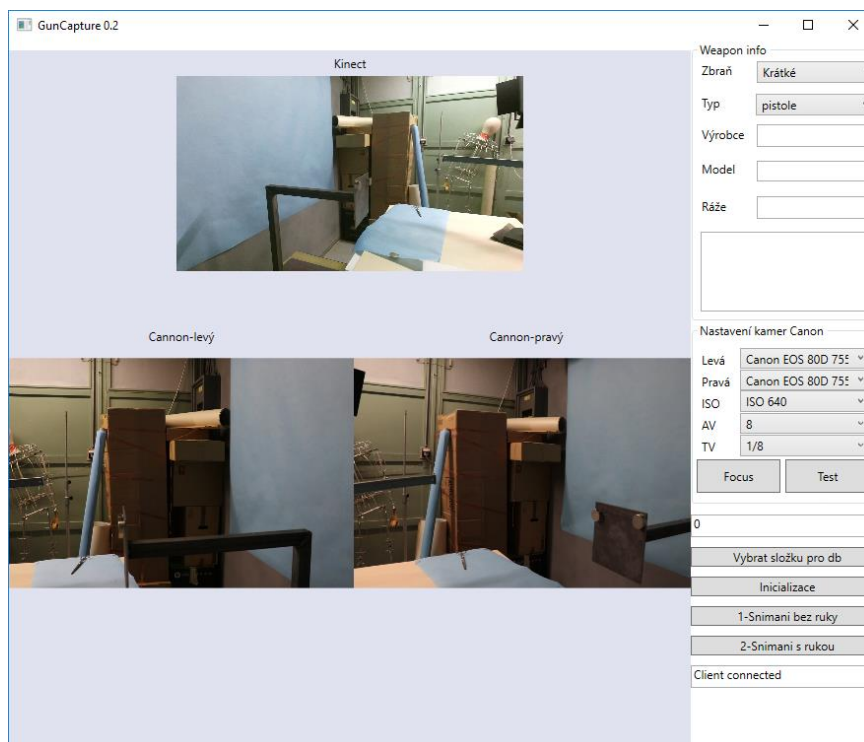
Aplikaci je zapotřebí po spuštění inicializovat – v průběhu inicializace dojde k vytvoření vlákna pro komunikaci se softwarem pro řízení vozíku. Dále se pak načtou data ze souhrnného souboru, která obsahují čísla adresářů a snímané typy zbraní. Tato funkce umožňuje navázat na předešlé snímání. Dále vytvoří aplikace XML soubor obsahující strukturované informace o zbraních, viz Kapitola 4. Dochází rovněž k ověření připojení fotoaparátů a k ustanovení komunikace s nimi. Následně aplikace vyčkává na spuštění snímání. Při inicializaci se nastaví i parametry pro zařízení Kinect. Jelikož aplikace umožňuje měnit nastavení fotoaparátu, tak se po inicializaci zobrazí živé náhledy z obou fotoaparátů. Hlavní výhodou této funkce je, že při změně nastavení můžeme okamžitě vidět, jak se změní výstupní obraz.

Pro zahájení snímání zašle aplikace softwaru pro řízení posunu snímačů příkaz k inicializaci. Po tomto příkazu se vozík U-rampy přesune do krajní polohy a aplikace (dle scénáře) může začít provádět snímání. Současně s inicializací vozíku se spustí vlákno s enkodérem, který vytváří z *Bitmap* videosekvenci ve formátu H.264. Snímání probíhá dle následujícího diagramu v Obrázku 8.



Obrázek 8 Diagram průběhu snímání.

Aplikace se chová jako server, ke které se připojuje software řídící vozík. Posun vozíku se provádí posláním příkazu přes TCP/IP. Druhá aplikace pak po dojezdu na místo pošle informaci o tom, že vozík se senzory dorazil do požadované pozice. Následně dojde k ustálení pohybu ramene s fotoaparáty, k vyfocení snímku a pořízení hloubkových dat. Jelikož se v každé pozici pořizují dva snímky, každý z nich se musí pro jednoznačné rozlišení ukládat do separátní složky (*left* nebo *right*). Stažení snímku probíhá ihned po jeho pořízení. Tímto způsobem probíhá snímání po celé trajektorii U-rampy, která je určena výšečí kružnice.



Obrázek 9 Ukázka uživatelského rozhraní aplikace pro pořizování dat.

4. Databáze

Cílem snímacího zařízení je vytvořit databázi s různými typy zbraní. Při vytvoření každé databáze je zapotřebí klást důraz na využitelnost. Z důvodu větší škály použití obsahuje databáze data v maximálním rozlišení, které poskytují jednotlivé senzory snímacího zařízení. Každá zbraň je přesně anotována v XML souboru, a to dle následujícího schématu (schéma bylo vytvořeno po konzultaci s Policií ČR):

Typ zbraně:

- krátká
 - o pistole
 - o revolver
- dlouhá
 - o jednoranové
 - o opakovací
 - o útočné
 - o samopaly

Další informace:

- výrobce
- model
- ráže

Adresářová struktura

- *dataset.txt* – soubor obsahující ID složek a názvy zbraní
- *XXX* – číselné automaticky generované ID zbraně (např. 001, 002,...)
 - *left* – složka pro fotografie z levého fotoaparátu

- *right* – složka pro fotografie z pravého fotoaparátu
- *pcl* – složka s PCL soubory
- *left_to_right.mp4* – videosoubor z přejezdu

Z hlediska dalšího zpracování bylo zapotřebí vytvořit seznam složek a k nim přiložit informaci o jaký typ zbraně se jedná. Tyto informace jsou obsažené v souboru *dataset.txt*.



Obrázek 10 Ukázka snímání zbraně do databáze.

5. Závěr

V rámci projektu byl vytvořen funkční vzorek (viz <http://www.fit.vutbr.cz/~drahan/prods.php?id=553¬itle=1>) zařízení umožňující provádět snímání zbraní v horizontální rovině v rozmezí 0-180°. Prostřednictvím zařízení můžeme vytvářet množinu snímků reprezentujících zbraň z různých úhlů. Výstupem je pak množina fotek, množina souborů s hloubkovými daty (PCL), videozáznam z přejezdu kolem zbraně a XML soubor obsahující informace o zbraní. Tato data mají široké možnosti využití v oblasti detekce a identifikace zbraně. V další fázi předpokládáme využití dat získaných prostřednictvím tohoto zařízení k vytvoření detektoru zbraní ve videozáznamu.

6. Literatura

- [1] KHOSHELHAM, Kourosh; ELBERINK, Sander Oude. Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors*, 2012, 12.2: 1437-1454.
- [2] WASENMÜLLER, Oliver; STRICKER, Didier. Comparison of kinect v1 and v2 depth images in terms of accuracy and precision. In: *Asian Conference on Computer Vision*. Springer, Cham, 2016. p. 34-45.
- [3] MICROSOFT Kinect hardware. In Kinect hardware [online]. [cit. 2018-01-14]. Dostupné na internetu: <https://developer.microsoft.com/en-us/windows/kinect/hardware>

[4] BILDSTEIN, J. Canon EDSK Tutorial in C#. In Canon EDSK Tutorial in C# - CodeProject[online]. [cit. 2018-01-14]. Dostupné na internetu: <https://www.codeproject.com/Articles/688276/Canon-EDSK-Tutorial-in-Csharp>