

**UNIVERZITA MATEJA BELA V BANSKEJ BYSTRICI**

**FAKULTA PRÍRODNÝCH VIED**

**IDENTIFIKÁCIA TEXTILNÝCH VLÁKIEN VO FORENZNOM  
SKÚMANÍ**

Diplomová práca

**0ca5e3cd-5c5f-48a4-89ef-7bf1b7acd65a**

Študijný program: Aplikovaná informatika

Študijný odbor: Aplikovaná informatika

Pracovisko (katedra, inštitút, ...): Katedra informatiky

Vedúci bakalárskej práce: Mgr. Michal Vagač, PhD.

**Banská Bystrica, 2019**

**Bc. Michaela Samuelčíková**



## ZADANIE ZÁVEREČNEJ PRÁCE

**Meno a priezvisko študenta:** Bc. Michaela Samuelčíková  
**Študijný program:** aplikovaná informatika (Jednoodborové štúdium, magisterský II. st., denná forma)  
**Študijný odbor:** aplikovaná informatika  
**Typ záverečnej práce:** Magisterská záverečná práca  
**Jazyk záverečnej práce:** slovenský  
**Sekundárny jazyk:** anglický

**Názov:** Identifikácia textilných vlákien vo forenznom skúmaní

**Anotácia:** Identifikácia vlákien predstavuje často dôležitý krok vo forenznom skúmaní. Na identifikáciu konkrétneho vlákna sú v súčasnosti používané mikroskopické metódy, pričom konkrétny nález sa porovnáva s atlasom známych vlákien. Alternatívnym postupom je napríklad infračervená a Ramanovská spektroskopia. Prvá metóda využíva mikroskopické vlastnosti daného vlákna, jeho tvar, štruktúru, pričom určenie druhu vlákna vyžaduje porovnanie so štandardom.  
Analyzujte možnosti klasifikácie mikroskopického obrazu textilného vlákna. Navrhňte a implementujte klasifikačný algoritmus a na reálnych dátach ho overte.

**Vedúci:** Mgr. Michal Vagač, PhD.  
**Katedra:** KIN FPV - Katedra informatiky  
**Vedúci katedry:** RNDr. Miroslav Melicherčík, PhD.  
**Dátum zadania:** 11.03.2018

**Dátum schválenia smerovej katedry:** 20.03.2018   
prof. RNDr. Roman Nedela, DrSc.  
garant študijného programu

### **Čestné vyhlásenie**

Čestne vyhlasujem, že som túto diplomovú prácu vypracovala samostatne s použitím uvedených zdrojov.

V Banskej Bystrici, 25.04.2019

.....

## **Abstrakt**

SAMUELČÍKOVÁ, Michaela, Bc.: Identifikácia textilných vlákien vo forenznom skúmaní.

[Diplomová práca] / Bc. Michaela Samuelčíková. – Univerzita Mateja Bela v Banskej Bystrici. Fakulta prírodných vied; Katedra informatiky. – Školiteľ: Mgr. Michal Vagač, PhD. Banská Bystrica FPV UMB, 2019, 66 s.

Práca sa zaoberá identifikáciou textilných vlákien vo forenznom skúmaní. V prvej časti je opísaná forezná chémia a jej využitie, rozdelenie vlákien a charakteristika jednotlivých tried ako aj metódy využívané na identifikáciu vlákien, z ktorých je bližšie priblížená spektroskopia. Venuje sa tiež neurónovým sieťam a ich podskupine konvolučných neurónových sietí.

Praktická časť sa venuje tvorbe modelu konvolučnej neurónovej siete na klasifikáciu vlákien. Na jeho tréning aj validáciu boli použité dáta z reálneho prostredia – fotografie vlákien pod mikroskopom.

**Kľúčové slová:** neurónová sieť, konvolučná neurónová sieť, spektroskopia, textilné vlákna, forezná veda

## **Abstrakt v anglickom jazyku**

SAMUELČÍKOVÁ, Michaela, Bc.: Identification of Textile Fibers in forensic investigation.

[Diploma thesis] / Bc. Michaela Samuelčíková. – Matej Bel University Banská Bystrica. Faculty of Natural Sciences; Department of computer sciences. – Supervisor: Mgr. Michal Vagač, PhD. Banská Bystrica FPV UMB, 2019, 66 p.

The work deals with the identification of textile fibers in forensic investigation. The first part describes the forensic chemistry and its utilization, the distribution of fibers and the characteristics of the individual classes as well as the methods used to identify the fibers from which the spectroscopy is explained closer. It also deals with neural networks and their subgroup of convolutional neural networks.

The practical part deals with making the model of the convolutional neural network for fiber classification. Real-world data was used to train and validate it - photographs of fibers under the microscope.

**Key words:** neural network, convolutional neural network, spectroscopy, textile fibers, forensic science

## **Predhovor**

S vyšetrovaním miest činu sa môžu spájať rôzne druhy stopových dôkazov. Stopové dôkazy bývajú malé a často sú tvorené iba mikroskopickými fragmentmi rôznych materiálov ako vlasy, vlákna z oblečenia alebo kobercov, úlomkami skla, kúskami pôdy alebo farby, ktoré sú spojené s tým, čo sa udialo počas páchania trestného činu, preto sa toho s ich pomocou dá veľa zistiť. Prenos stopových materiálov môže spojiť podozrivého, obeť a miesta činu a preto môžu byť použité na poskytnutie rozhodujúcich dôkazov v kriminálnych prípadoch. Najčastejším využitím je poskytnutie dôkazu o možnom kontakte medzi osobami podozrivými z trestných činov ako je vražda, znásilnenie, sexuálny útok, útok so zbraňou alebo útok, pri ktorom došlo k ťažkému ublíženiu na zdraví, poprípade preukázanie kontaktu medzi osobami a inými textilnými povrchmi ako napríklad sedadlá v aute alebo poťahy na nábytku. Môžu byť tiež využité pri vyšetrovaní automobilových nehôd, pri ktorých vodič z miesta nehody ušiel, alebo môžu pomôcť pri dokazovaní šoférovania ukradnutého vozidla či šoférovania pod vplyvom alkoholu.

V tejto práci sa spracúva téma identifikácie textilných vlákien vo forenznom skúmaní. Sú tu priblížené vlákna – ich rozdelenie a charakteristika a tiež rôzne metódy prístupu ich identifikácie, medzi inými aj použitím fotografií vlákien pod mikroskopom a ich následnou klasifikáciou konvulučnou neurónovou sieťou - triedou neurónových sietí zameranou na rozpoznávanie a klasifikáciu obrázkov, detekciu objektov a rozpoznávanie tvárí s podstatným využitím v oblasti počítačového videnia.

Cieľom práce bolo vytvoriť model neurónovej siete, ktorý by vedel priradiť hľadané vlákno do jednej z tried vlákien využitých pri vytváraní tohto modelu.

## **PodĎakovanie**

Chcela by som poďakovať Mgr. Michalovi Vagačovi, PhD., vedúcemu mojej diplomovej práce, za jeho pomoc, čas, rady a pripomienky, ktoré mi pri písaní práce poskytol. Ďalej by som sa chcela poďakovať Mgr. Michalovi Povinskému za jeho pomoc, RNDr. Šimonovi Budzákovi, PhD. a Viktórii Živickej za pomoc a poskytnutie materiálov a Katedre chémie a Katedre biológie Fakulty prírodných vied Univerzity Mateja Bela v Banskej Bystrici za zapožičanie vybavenia.

# Obsah

Úvod .....	10
<b>1 Identifikácia textilných vlákien vo forenznom skúmaní .....</b>	<b>11</b>
<b>1.1 Forezná veda a jej využitie .....</b>	<b>11</b>
<b>1.2 Metódy na identifikáciu vlákien .....</b>	<b>11</b>
1.2.1 Netechnické metódy .....	12
1.2.2 Technické metódy .....	12
<b>1.3 Druhy vlákien .....</b>	<b>13</b>
1.3.1 Prírodné vlákna .....	13
1.3.2 Syntetické vlákna .....	21
<b>1.4 Spektroskopia .....</b>	<b>28</b>
1.4.1 Infračervená spektroskopia .....	28
1.4.2 Ramanova spektroskopia .....	29
<b>1.5 Neurónová sieť .....</b>	<b>31</b>
1.5.1 Neurón .....	31
1.5.2 Aktivačné funkcie .....	32
1.5.3 Učenie .....	32
1.5.4 Architektúra neurónových sietí .....	34
1.5.5 Algoritmus spätného šírenia chyby .....	34
<b>1.6 Konvolučná neurónová sieť .....</b>	<b>35</b>
<b>2 Návrh a implementácia metódy na identifikáciu vlákien vo forenznom skúmaní</b>	<b>38</b>
<b>2.1 Príprava dát .....</b>	<b>38</b>
<b>2.2 Príprava dát pre neurónovú sieť .....</b>	<b>38</b>
2.2.1 Vytvorenie menších výrezov vlákna .....	39
2.2.2 Úprava výrezov .....	50

<b>2.3</b>	<b>Neurónová sieť</b> .....	<b>53</b>
2.3.1	Definovanie konvolučnej neurónovej siete.....	53
2.3.2	Algoritmus neurónovej siete .....	58
2.3.3	Módy neurónovej siete.....	60
	<b>Záver</b> .....	<b>61</b>
	<b>Zoznam bibliografických odkazov</b> .....	<b>63</b>
	<b>Prílohy</b> .....	<b>67</b>
	<b>Príloha A – Používateľská príručka</b> .....	<b>67</b>
	<b>Príloha B – Systémová dokumentácia</b> .....	<b>72</b>



## Zoznam skratiek a značiek

EVA	Etylenvinylacetát
PAN	Polyakrylnitril
PE	Polyetylen
PET	Polyetylentereftalát
PP	Polypropylén
OpenCV	Open Source Computer Vision Library
PIL (Pillow)	Python Imaging Library

## Úvod

Forenzné dôkazy vo forme vlákien hrajú dôležitú úlohu v množstve kriminálnych prípadov. Kedysi využívané metódy, ktoré poškodzovali vlákna, začali nahrádzať nedeštruktívne metódy, ktoré dávajú presnejšie výsledky a vlákno uchovávajú v nepoškodenom stave. Nedeštruktívne techniky, ktoré dokážu buď rozlišovať medzi podobnými vláknami alebo spojiť hľadané vlákno so známym vláknom a stále pri tom zachovať integritu vlákien pre ďalšie súdne vyšetrenie sú vo forenznej vede veľmi cenné. Medzi takéto techniky patrí aj spektroskopia - štúdium vzájomného pôsobenia medzi hmotou a elektromagnetickým žiarením. Vo forenznej vede je najviac využívaná infračervená spektroskopia založená na meraniach infračerveného žiarenia absorbovaného alebo odrazeného od molekúl skúmanej vzorky a Ramanova spektroskopia, ktorá skúma nepružný rozptyl monochromatického svetla odrazeného od molekúl skúmanej vzorky.

V teoretickej časti sa venujeme forenznej vede, jej využitiu a tiež ostatným metódam na identifikáciu textilných vlákien vo foreznom skúmaní, ktoré zahŕňajú ako metódy voči vláknam nedeštruktívne – ku ktorým patrí napríklad spektroskopia, tak aj deštruktívne. Opísané je rozdelenie vlákien a jednotlivé triedy vlákien, s ktorými pracujeme v praktickej časti a ich vlastnosti a správanie pri použití týchto metód. Tiež sa venujeme konvolučným neurónovým sieťam, ktoré sú využívané aj pri praktickej časti a neurónovým sieťam a ich princípu vo všeobecnosti.

V praktickej časti tejto práce je implementovaná jedna z nedeštruktívnych techník, pri ktorej sú využívané fotografie vlákien pod mikroskopom, ktoré boli spracované a poskytnuté konvolučnej neurónovej sieti ako klasifikačná vzorka za účelom vytvorenia modelu na identifikáciu známych vlákien.

# **1 Identifikácia textilných vlákien vo forenznom skúmaní**

Rovnako ako vlasy, patria textilné vlákna medzi najbežnejšie veci zanechané na mieste činu. Pretože sú ale textílie masovo vyrábané, je ťažké vysledovať vlákno späť k určitému zdroju. Napriek tomu sú dôkazy v podobe vlákien cenné, pretože vytvárajú väzby medzi miestami, obeťami a podozrivými, a čokoľvek, čo môže pomôcť zúžiť pôvod vlákna na obmedzený počet zdrojov, robí toto vlákno omnoho cennejším dôkazom. Pri identifikácii vlákien porovnávajú vyšetrovatelia fyzické, chemické a mikroskopické vlastnosti vlákna z miesta činu a známeho vlákna, pričom zvyčajne musia vykonať veľa testov na identifikáciu a porovnanie dôkazových vlákien [35].

## **1.1 Forezná veda a jej využitie**

Oblasť foreznej vedy čerpá z mnohých vedných odborov ako sú fyzika, chémia a biológia, pričom sa zameriava na rozpoznanie, identifikáciu a hodnotenie fyzických dôkazov. Stala sa dôležitou súčasťou súdneho systému, pretože využíva široké spektrum vedy na získanie informácií týkajúcich sa trestnoprávných dôkazov.

Forezná veda môže dokázať existenciu trestného činu, páchatel'a tohto trestného činu alebo jeho spojenie s trestným činom prostredníctvom preskúmania fyzických dôkazov, vykonávania testov a interpretácie údajov. Stala sa neoddeliteľnou súčasťou mnohých trestných prípadov a odsúdení, pričom objektívne fakty vo forme vedeckých poznatkov slúžia obom stranám – ako obhajobe, tak aj obžalobe. Svedectvo forezných vedcov sa stalo dôveryhodnou súčasťou v mnohých občianskych a trestných prípadoch, keďže títo odborníci sa nezaoberajú výsledkom prípadov, iba ich objektívnym svedectvom založeným čisto na vedeckých faktoch [2], [3], [35].

## **1.2 Metódy na identifikáciu vlákien**

Forezní vedci vykonávajú ako fyzikálnu, tak aj chemickú analýzu na fyzických dôkazoch získaných vyšetrovateľmi z miesta činu. Títo vední experti používajú mikroskopické vyšetrovacie techniky, komplexné nástroje, matematické a vedecké princípy a referenčnú literatúru na analýzu dôkazov a na identifikáciu triedových a individuálnych charakteristík. Základné rozdelenie týchto techník a metód je na metódy technické a na metódy netechnické [1].

### **1.2.1 Netechnické metódy**

Netechnické metódy si nevyžadujú žiadne špeciálne vybavenie a pomôcky. Na identifikáciu vlákna sa používajú testy, pri ktorých je vlákno skúmané dotykom alebo je skúmané správanie vlákna vystaveného priamemu kontaktu s ohňom.

#### **1.2.1.1 Test dotykom**

Pri tomto teste skúma odborník látku hmatom a zrakom. Stať sa takýmto odborníkom si vyžaduje roky praxe a manipuláciu s veľkým množstvom rozličných látok a vlákien. Obmedzením tejto metódy je porovnávanie látok vyrobených z viacerých druhov vlákien a tiež schopnosti odborníka. Využíva sa, medzi iným, aj tepelná vodivosť látky. Napríklad vlnená látka je na dotyk teplá, pretože vlna nie je vodičom tepla a teda teplo z dotyku zostane v mieste dotyku. Naopak, látky vyrobené z rastlinných vlákien ako bavlna, ľan alebo hodváb sú dobrými vodičmi tepla, teplo z dotyku odvedú a preto na dotyk pôsobia chladne.

#### **1.2.1.2 Test horením**

Pri tomto teste je spálený ústrižok látky, prípadne vlákna a pozorujú sa jeho vlastnosti pri horení. Je účinnejší ako test dotykom, ale taktiež pri ňom nastávajú komplikácie pri zmesových látkach alebo kvôli chemickým povrchovým úpravám, ktoré menia vlastnosti látok. Hlavnou nevýhodou je znehodnotenie vlákna alebo látky použitej na test. K pozorovaným vlastnostiam patrí farba ohňa, farba a vôňa dymu, popol alebo spôsob horenia. Väčšina tkanín horí a musia byť uhasené, ale existujú aj také, ktoré zhoria úplne, také, ktoré horia, ale po chvíli sa oheň sám uhasí, pričom zanechá zvyšok nezhorenej látky, alebo také, ktoré nehoria vôbec.

### **1.2.2 Technické metódy**

Technické metódy si vyžadujú špeciálne vybavenie a/alebo špeciálne pomôcky ako napríklad mikroskop ale špecializované laboratóriá.

#### **1.2.2.1 Test pod mikroskopom**

Tento test na identifikáciu vlákien využíva mikroskop s priblížením minimálne 100 jednotiek, vďaka čomu dokáže vlákna rozpoznať s veľkou úspešnosťou. Ľahšie však rozpoznáva prírodné vlákna ako syntetické, ktoré majú navzájom veľmi podobnú štruktúru.

### **1.2.2.2 Chemické testy**

Tieto testy sa uskutočňujú v špecializovaných laboratóriách a poskytujú presnú analýzu vlákien. Stávajú sa ale menej využívanými, pretože väčšina syntetických vlákien a ich zmesí má podobnú chemickú štruktúru a tiež neexistuje žiaden chemický test na rozlíšenie jednotlivých vlákien v zmesiach.

#### **Test farieb**

Je založený na teórii, že každé vlákno má odlišnú dvojfarebnú kombináciu ako reakciu na vystavenie kyseline a zásade. Vlákno sa zafarbí špecifickou farbou, keď je vystavené zriedenej kyseline octovej a inou špecifickou farbou, keď je vystavené slabej zásade.

#### **Test rozpúšťadiel**

Identifikuje vlákna na základe ich skúmaných reakcií pri vystavení rôznym rozpúšťadlám.

## **1.3 Druhy vlákien**

Vlákna sa rozdeľujú do dvoch hlavných kategórií a to na prírodné a syntetické. V nasledujúcej kapitole sú opísané niektoré triedy vlákien, ich vlastnosti a ich reakcie na metódy identifikácie vlákien čerpané z [3], [6], [13], [14], [15].

### **1.3.1 Prírodné vlákna**

Prírodné vlákna sa ďalej delia na vlákna rastlinného a vlákna živočíšneho pôvodu.

#### **1.3.1.1 Rastlinné vlákna**

##### **Bavlna**

Vlákna bavlny sa získavajú zberom vlákien rastúcich v kapsuliach okolo semien bavlníka. Sú zväčša tvorené celulórou a sú hydrofilné.

Na pohľad sú biele a väčšinou matné, bez lesku. Majú dobrú tepelnú vodivosť, sú mäkké a málo pružné.

Pri kontakte s ohňom vzbĺknu a horia, pričom vydávajú zápach zhorených listov alebo papiera.

Sú odolné voči zásadám, acetónu a organickým rozpúšťadlám, ale neodolné voči koncentrovaným kyselinám a zahriatym zriedeným kyselinám.

Využívajú sa hlavne pri výrobe textilu a sú dnes prítomné vo viac ako 50 % vyrábaných textílií ako sú tričká, ponožky, spodná bielizeň, ale tiež uteráky, župany, posteľné obliečky,

používajú sa pri výrobe rifl'oviny, menčestroviny, vlny na štrikovanie a vyrábajú sa z nich aj rybárske siete, kávové filtre a stany.



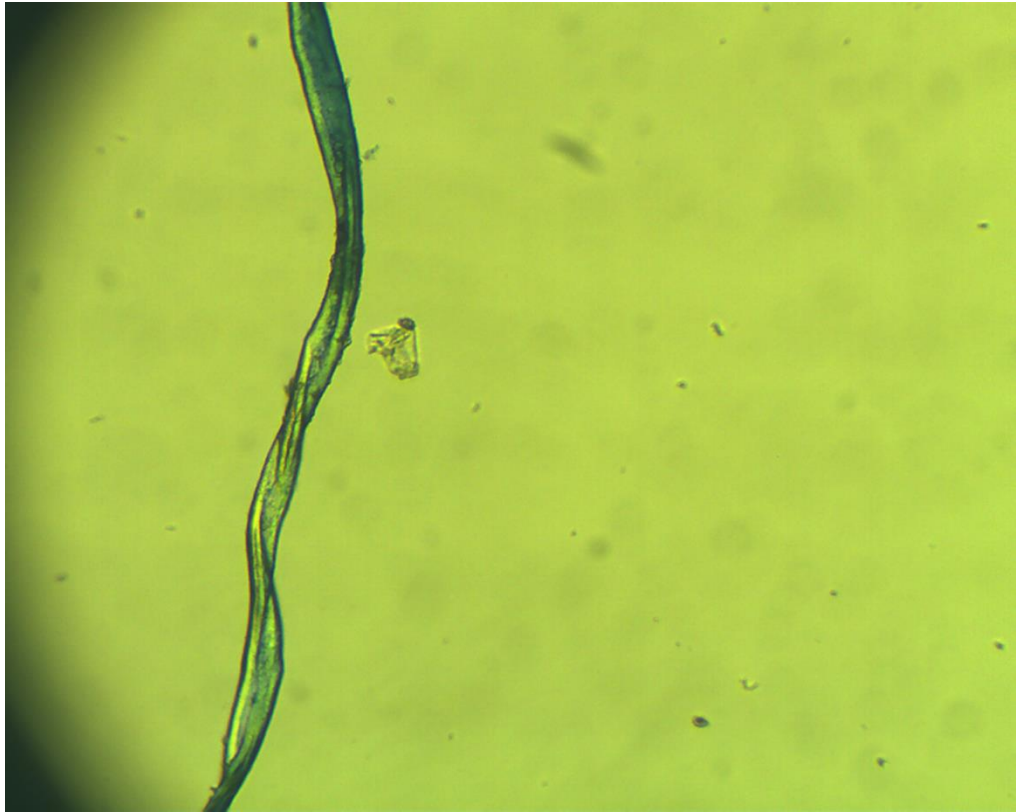
Obrázok 1 - fotografia vlákna bavlny pod mikroskopom (zdroj: autor)

### **Rifl'ovina**

Rifl'ovina je pevná a trvanlivá bavlnená tkanina utkaná keprovou väzbou, pri ktorej sa striedajú dolné a horné nite tak, aby horné nite vytvárali na povrchu látky sústavu šikmých súbežných riadkov. Ďalej sa najčastejšie farbí indigovou farbou, pričom zafarbené sú iba vrchné vlákna a spodné zostávajú biele.

Keďže je to bavlnená tkanina, majú jej vlákna rovnaké vlastnosti ako je dobrá tepelná vodivosť a nízka pružnosť. Pri kontakte s plameňom vzbĺknu a horia a sú odolné voči zásadám, acetónu a organickým rozpúšťadlám, ale neodolné voči koncentrovaným kyselinám a zahriatym zriedeným kyselinám.

Rifl'ovina sa využíva na výrobu čižiem, nohavíc, šiat, bünd, tričiek, košiel, opaskov, kabeliek, tienidiel na lampy a poťahov na nábytok.



Obrázok 2 - fotografia vlákna riff'oviny pod mikroskopom (zdroj: autor)

### **Juta**

Vlákná juty sa získavajú z rastliny jutovníka, skladajú sa predovšetkým z rastlinných materiálov ako celulóza a lignín a používajú sa na tvorbu látky zvanej jutovina.

Vlákná sú dlhé, jemné a lesklé, kvalitnejšie spracované vlákna by mali mať žltú, zlatú až červenkastú farbu, menej kvalitne spracované vlákna môžu byť hnedé až zelené.

Pri kontakte s ohňom juta rýchlo vzplanie a horí, pričom vydáva zápach spáleného papiera alebo dreva.

Je neodolná voči koncentrovaným kyselinám, zahriatym zriedeným kyselinám a silným zásadám.

Používa sa predovšetkým na výrobu vriec, tašiek, podložiek, bytového textilu a kobercov.



**Obrázok 3 - fotografia vlákna juty pod mikroskopom (zdroj: autor)**

### **Lan**

Lanové vlákna sa získavajú z lyka stromu ľanu siateho, sú hydrofilné a silné.

Vlákna sú lesklé, béžovej, hnedej alebo sivej farby. Po silnom bielení môžu dosiahnuť biely odtieň. Na dotyk sú chladné, jemné a neelastické.

Po vystavení priamemu ohňu vzbĺknu rýchlo a horiace sú ľahko uhasiteľné, pričom vydávajú zápach horiacich listov alebo dreva.

Sú odolné voči zriedeným kyselinám (ak sú ihneď po aplikácii umyté) a organickým rozpúšťadlám. Voči koncentrovaným kyselinám odolné nie sú.

Používajú sa na výrobu letného vrchného ošatenia a obuvi, maliarskych plátien, prestieradiel, obrusov, utierok, dekorácií a väzby kníh.

#### **1.3.1.2 Živočíšne vlákna**

### **Vlna**

Vlna je textilné vlákno získané z vlny oviec, kôz, pižmoňov, bizónov, zajacov a iných zvierat. Vonkajšia časť vlákna je hydrofóbna a vnútorná hydrofilná.



Vlákná sú na pohľad lesklé, bielej, hnedej alebo čiernej farby. Sú tiež zvlnené a elastické.

Pri vystavení priamemu ohňu sa vlákna od plameňa odťahujú, horia pomaly a po zamedzení kontaktu s plameňom horieť prestanú. Počas horenia vydávajú zápach horiaceho peria alebo vlasov.

Vlákná vlny sú odolné voči minerálnym kyselinám, slabým zásadám a organickým rozpúšťadlám. Voči ostatným kyselinám a zásadám odolné nie sú.

Vlna sa využíva pri výrobe oblečenia, diek, spodnej bielizne, látkových plienok, diek pod sedlo, kobercov, izolácie, čalúnenia a tiež na absorbovanie hluku v stereo súpravách.

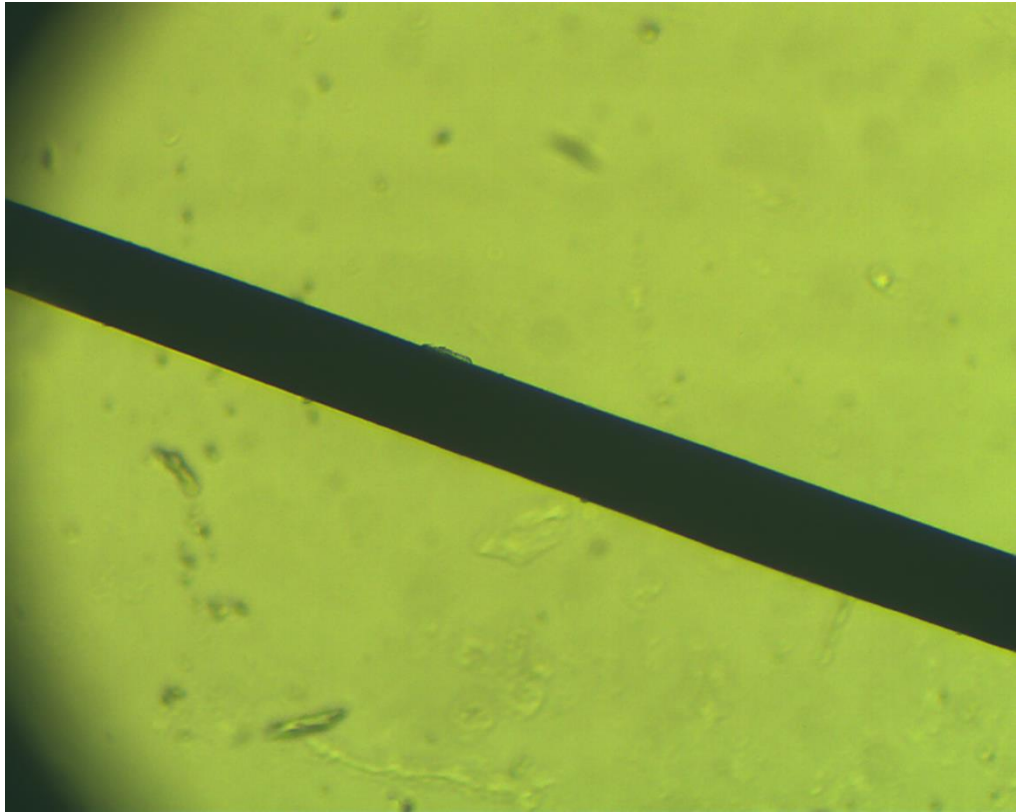


**Obrázok 4 - fotografia vlákna vlny pod mikroskopom (zdroj: autor)**

### **Srst'**

Srst' označuje hustý porast chlupov zakrývajúcich kožu zvierat – cicavcov. Je hydrofóbná a teda odpudzuje vodu. Jej farba aj štruktúra závisí od druhu zvierat'a. Pri kontakte s priamym ohňom sa od plameňa odťahuje, horí pomaly a po zamedzení kontaktu s plameňom horieť prestane. Počas horenia vydáva zápach horiaceho peria alebo vlasov.

V textilnom priemysle sa využíva na výrobu topánok, bünd, kabátov a tiež pri výrobe plsti.



**Obrázok 5 - fotografia srsti mačky pod mikroskopom (zdroj: autor)**



**Obrázok 6 - fotografia srsti zajaca pod mikroskopom (zdroj: autor)**

## Velúr

Označenie velúr sa používa pre rôzne druhy textílií s krátkym vlasom, ktoré sa vyznačujú zamatovým jemne chlpatým povrchom a používajú sa ako napodobenina zamatu.

Môže predstavovať plyšovú tkanú látku alebo textíliu vyrobenú zo zdrsnených, česaných, často pristrihovaných bavlnených, vlnených alebo tiež syntetických (napríklad polyester) vlákien. Táto textília je elastická a jemná, využíva sa na výrobu oblečenia, čalúnenia, poťahov na sedadlá v aute, trikotov, šiat a javiskových opôn a závesov. Fyzikálne a chemické vlastnosti preberá od druhu vlákna, z ktorého je vyrobená.

Velúr tiež môže označovať látku vyrobenú zo spodnej strany najčastejšie teľacej alebo hovädzej kože s jemným vlasom, tiež nazývanú semiš. Semiš je hydrofílny a menej trvácny a jemnejší ako koža.

Na dotyk je jemný, tenký a ohybný. Po vystavení priamemu ohňu horí, ale po zamedzení kontaktu s plameňom horieť prestane.

Pôvodne sa využíval na výrobu dámskych rukavíc, ale dnes sa z neho vyrába aj čalúnenie, topánky a kabelky.



Obrázok 7 - fotografia vlákna velúru pod mikroskopom (zdroj: autor)

## Hodváb

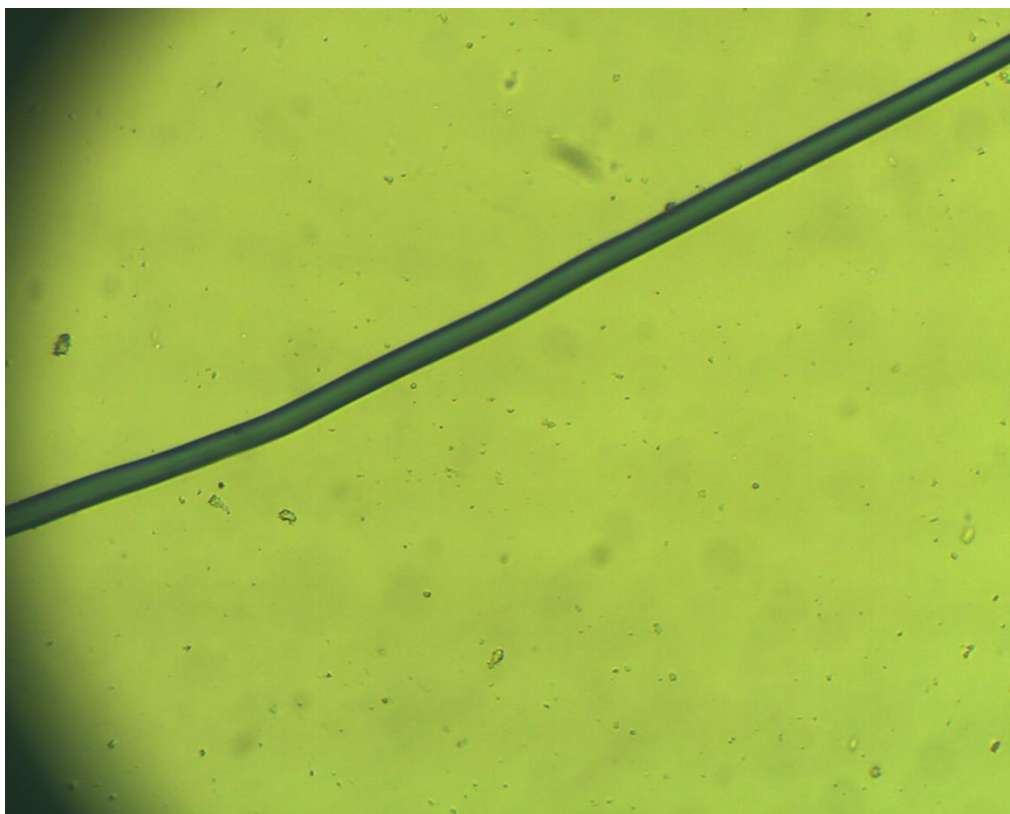
Hodváb je prírodné proteínové vlákno, ktoré je produkované larvami špecifických druhov hmyzu na vytvorenie zámotkov. Hodváb dokáže vytvárať niekoľko druhov hmyzu, ale na výrobu textílií sa využíva iba hodváb produkovaný húsenicami priadky morušovej.

Hodvábne vlákna sú lesklé, žltej, hnedej, zelenej alebo sivej farby. Vďaka svojej štruktúre lámucej svetlo dokážu ale odrážať svetlo pod rôznymi uhlami a vytvárať tak rôzne farby. Na dotyk sú hladké, mäkké, chladné, neelastické, ale nie klzké.

Pri vystavení priamemu ohňu sa vlákno odťahuje od plameňa, horí pomaly a po zamedzení kontaktu s plameňom horieť prestane. Pri horení vydáva zápach spáleného peria, vlasov alebo mäsa.

Je odolný voči slabým zriedeným kyselinám, minerálnym kyselinám (okrem kyseliny sírovej), slabým zásadám a organickým rozpúšťadlám. Voči ostatným kyselinám, zásadám a chlóróvemu bielidlu odolný nie je.

Používa sa na výrobu tričiek, kravát, blúzok, formálnych šiat, spodnej bielizne, pyžám a županov.



Obrázok 8 - fotografia vlákna hodvábu pod mikroskopom (zdroj: autor)

### 1.3.2 Syntetické vlákna

#### Celofán

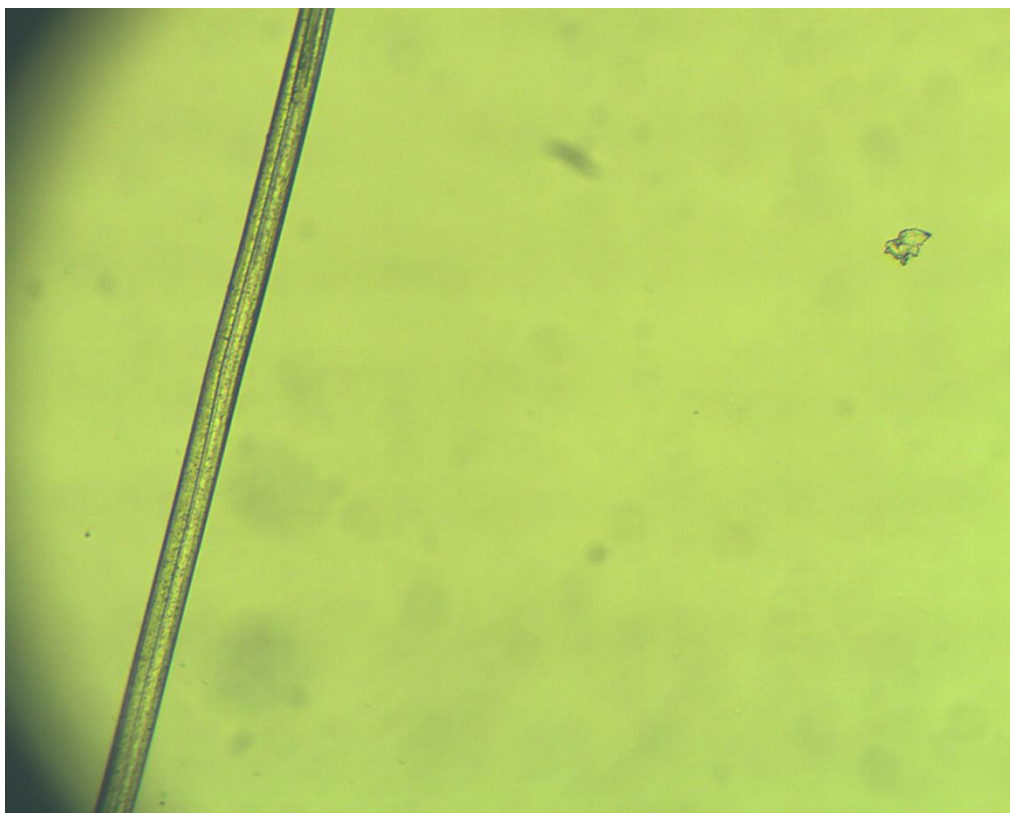
Celofán sa vyrába chemickými úpravami z regenerovanej celulózy získanej z dreva, bavlny, konope a iných látok. Je nepriepustný pre vodu, vzduch, oleje a baktérie, ale je vysoko priepustný pre vodné pary.

Surový materiál je biely ako bavlna, spracovaný je priehľadný a umiestnený medzi dva polarizačné filtre rozkladá svetlo ako sklenený hranol.

Surový materiál horí prudko a vydáva zápach spálených listov, spracovaný sa pri kontakte s plameňom zmršťuje.

Je odolný voči kyselinám, zásadám, tukom a olejom, alkoholom, uhl'ovodíkom, halogénovým uhl'ovodíkom, ketónom a esterom. Nie je odolný voči koncentrovaným kyselinám.

Vyrábajú sa z neho predovšetkým fólie a obaly na jedlo, ale tiež lepiaca páska, hadice a obaly na cigary.



Obrázok 9 - fotografia vlákna celofánu pod mikroskopom (zdroj: autor)

### **Etylvinylacetát (EVA)**

Etylvinylacetát alebo polyetylénvinylacetát (PEVA) je termoplastický polymér. Má podobné vlastnosti ako polyetylén, ale má väčší lesk, mäkkosť a flexibilitu, taktiež má niektoré vlastnosti gummy. Je vodeodolný, odolný voči UV žiareniu a má dobrú flexibilitu aj pri nízkych teplotách (až do  $-70\text{ }^{\circ}\text{C}$ ).

Má octový zápach, v normálnom stave je priehľadný.

Pri kontakte s priamym ohňom sa tavia.

Je odolný voči kyselinám, zásadám, tukom a olejom, alkoholom a alifatickým uhlíkovodíkom. Má slabú odolnosť voči aromatickým uhlíkovodíkom ako toluén alebo xylén.

Používa sa ako výplň do lyžiarskych a wakeboardových topánok, sedla na bicykli, hokejových vypchávk, rukavíc a heliem na box a zmiešané bojové umenia, do rybárskych prútov. Ako lisovaná pena sa využíva tiež na tlmenie nárazov v športových a bežeckých topánkach.



**Obrázok 10 - fotografia vlákna etylvinylacetátu pod mikroskopom (zdroj: autor)**

## Nylon

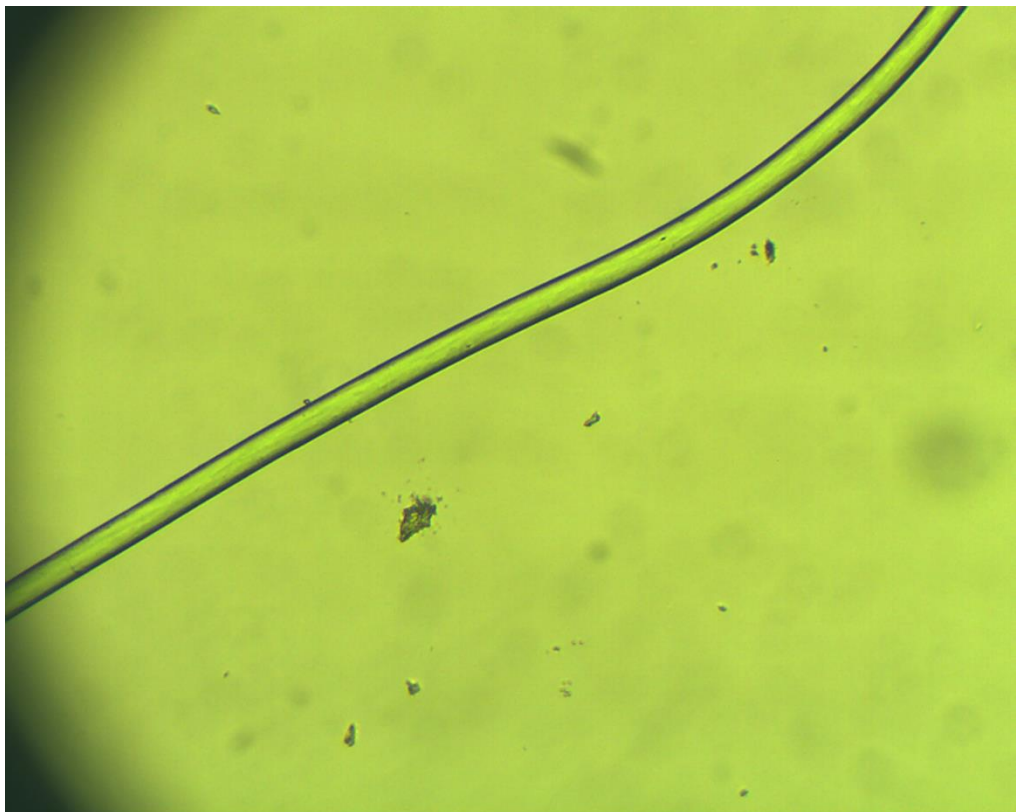
Nylon je označenie pre triedu syntetických polymérov, často označovaných ako polyamidy. Názov pôvodne označoval skupinu polyamidov 6.6, ale využíva sa tiež na označenie všetkých polyamidov. Je to termoplastický kondenzačný polymér. Jeho najčastejšie sa vyskytujúcimi typmi sú Nylon 6 a Nylon 6.6, ktoré sa líšia predovšetkým v chemickej štruktúre. Zatiaľ čo Nylon 6 je flexibilnejší a využíva sa napríklad na štetiny zubných kefiek, Nylon 6.6 je trvácnejší a odolnejší a je možné ho nájsť napríklad v kobercoch.

Na dotyk je podobný hodvábu a elastický.

Horí modrým plameňom so žltým vrcholom a pri horení vydáva zápach zhorených vlasov alebo vlny.

Je odolný voči alkoholom, zásadám, tukom, olejom, ketónom a aromatickým uhl'ovodíkom. Rozpúšťa sa v kyselinách a halogénoch.

Využíva sa pri výrobe pančuchových nohavíc, športových odevov, padákov, sietí, výpletov tenisových rakiet, rybárskych laniek, povrazov a lán.



Obrázok 11 - fotografia vlákna nylonu pod mikroskopom (zdroj: autor)

### **Polyakrylnitril (PAN)**

Polyakrylnitril je termoplastický syntetický polokryštalický organický polymér.

Vlákno je na dotyk pružné a mäkké.

Topí sa pri teplote nad 300°C.

Je odolný voči kyselinám, zásadám, tukom, olejom a alkoholom. Rozpúšťa sa v aromatických uhl'ovodíkoch, halogénoch a ketónoch.

Používa sa na výrobu filtračných membrán, tkanín, textílií, prikrývok, poťahov nábytku a imitácií kožušín.



**Obrázok 12 - fotografia vlákna polyakrylnitrilu pod mikroskopom (zdroj: autor)**

### **Polyetylen (PE)**

Polyetylén je najčastejšie sa vyskytujúcim plastom. Je to termoplast, ktorý vzniká polymerizáciou etylénu a väčšinou je tvorený zmesou jeho podobných polymérov.

Môže byť priehľadný, priesvitný alebo nepriehľadný. Nefarbený je väčšinou matný, na dotyk voskovitý a ľahko poškrabateľný.

Horí modrým plameňom so žltým vrcholom a po odstránení zdroja ohňa horieť neprestane.



Je odolný voči silným kyselinám a zásadám, redukčným činidlám a slabým oxidantom. Rozpúšťa sa v aromatických uhľovodíkoch ako toluén, xylén alebo v chlórových rozpúšťadlách ako trichlóretán a trichlórbenzén. Neabsorbuje takmer žiadnu vodu, jeho priepustnosť plynov a vodnej pary je nízka, naopak má vysokú priepustnosť kyslíku, oxidu uhličitého a aromatických zlúčenín.

Využíva sa na výrobu fólií, vriec na odpad, implantátov a protéz alebo na šitie v chirurgii.



Obrázok 13 - fotografia vlákna polyetylenu pod mikroskopom (zdroj: autor)

### **Polyetylentereftalát (PET)**

Polyetylentereftalát je termoplastický polymér a radí sa do skupiny so súhrnným označením polyester (PES). Z tejto triedy je najbežnejšie sa vyskytujúcim polymérom. Pripravuje sa polykondenzačnou reakciou a existuje v dvoch verziách – ako amorfný (priehľadný) alebo polykryštalický polymér.

V bežnom stave je bezfarebný a odolný voči zafarbeniu (na farbenie polyesterov sa používajú špeciálne rozptyľové farbivá). Vlákna sú lesklé a na dotyk hladké a jemné.

Je hydrofilný, neprepúšťa plyny, pary ani vlhkosť, po úprave neprepúšťa alkoholy ani ďalšie rozpúšťadlá. Po vystavení chloroformu a niektorým iným chemikáliám (ako toluén) zbelie.

Využíva sa na výrobu oblečenia, posteľného povlečenia, diek, potáhov a nábytku, ako výplň vankúšov ale aj pri výrobe bezpečnostných pásov v autách, na výrobu kobercov, ako vlákno do 3D tlačiarň a zväzuje sa do lán držiacich zvony.



Obrázok 14 - fotografia vlákna polyetyltereftalátu pod mikroskopom (zdroj: autor)

### **Polypropylén (PP)**

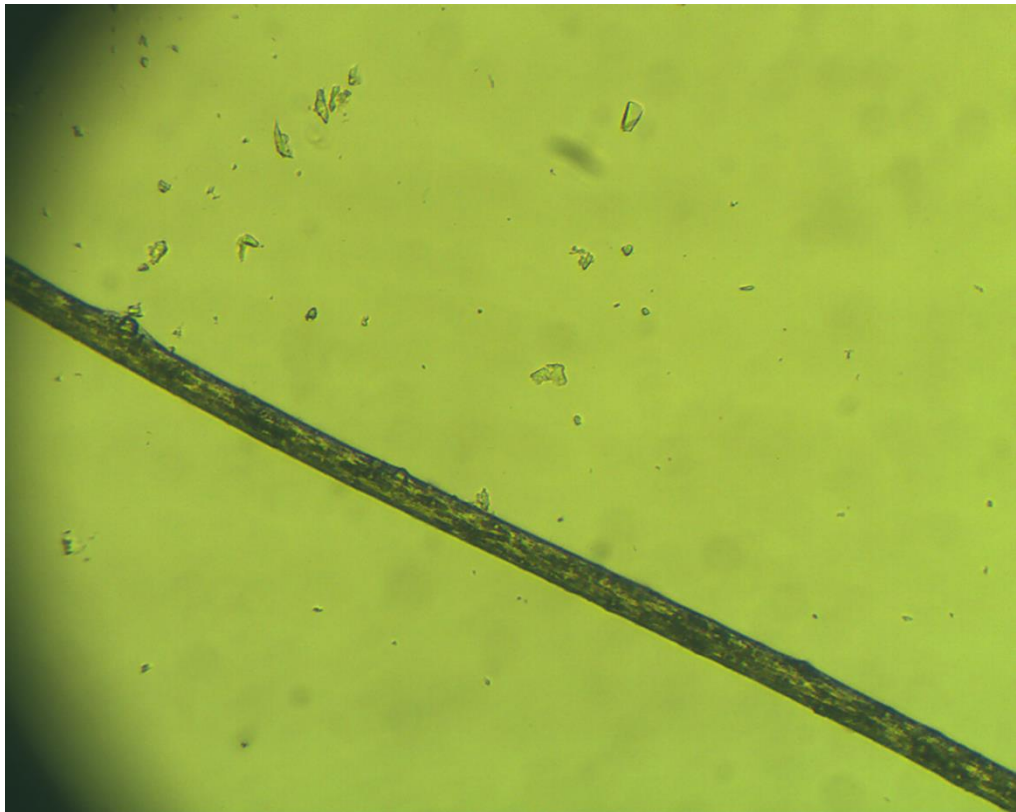
Polypropylén je čiastočne kryštalický termoplastický polymér v mnohom podobný polyetylénu. Získava sa adičnou polymerizáciou propénu a z jeho troch známych variant (izotaktické polyméry, syndiotaktické polyméry, ataktické polyméry) sa na výrobu textilných vlákien využívajú iba izotaktické polyméry, ktoré majú rovnakú molekulárnu štruktúru. Je hydrofilný, má nízku hustotu a je najľahším zo všetkých vlákien.

Polypropylén je biely drsný materiál, ktorý sa na dotyk nelíši od ovčej vlny. Vlákna môžu byť priehľadné alebo inak zafarbené.

Nie je ľahko horľavý a pri vystavení ohňu sa topí.

Má veľmi dobrú chemickú a mechanickú odolnosť, jeho chemická odolnosť závisí ale od teploty, pretože pri nízkych teplotách krehne. Pri izbovej teplote je odolný voči tukom, alkoholom, olejom, uhl'ovodíkom, aldehydom, zásadám a takmer všetkým organickým rozpúšťadlám, okrem silných kyselín. Dobre sa rozpúšťa v xyléne, tetrahydronaftaléne a dekalíne. Má slabú až žiadnu odolnosť voči halogénom a oxidačným činidlám.

Využíva sa v netkaných látkach na výrobu plienok, hygienických produktov, vzduchových a plynových filtrov, filtrov na vodu a tekutiny a tiež na olej zachytávajúce filtre v plávajúcich bariérach na odstraňovanie olejových škvŕn. Z oblečenia sa z neho vyrábajú spodné vrstvy oblečenia do chladného počasia ako trička s dlhým rukávom alebo dlhá spodná bielizeň.



**Obrázok 15 - fotografia vlákna polypropylénu pod mikroskopom (zdroj: autor)**

## 1.4 Spektroskopia

Svetlo s hmotou na seba vzájomne pôsobia rôznymi spôsobmi. Niektorými materiálmi svetlo prechádza, od iných sa odráža alebo rozptyľuje. Túto interakciu ovplyvňuje ako materiál, tak farba svetla (vlnová dĺžka svetla). Štúdium tohto svetla sa nazýva spektroskopia. Spektroskopia teda študuje vzájomné pôsobenie medzi hmotou a elektromagnetickým žiarením. Jednými z najpoužívanejších metód spektroskopie vo forenznom skúmaní sú infračervená spektroskopia a Ramanova spektroskopia.

### 1.4.1 Infračervená spektroskopia

Infračervená spektroskopia patrí do skupiny nedeštruktívnych analytických metód, pri ktorej nie je skúmaná vzorka poškodená a napriek tomu poskytuje informáciu o svojom zložení. Patrí do oboru molekulovej spektroskopie a napriek tomu, že sa využíva pri analýze makroskopických vzoriek, študuje vzájomné pôsobenie infračerveného žiarenia s mikročasticami (molekulami) tejto vzorky [28]. Využíva fakt, že molekuly absorbujú frekvencie, ktoré sú charakteristické pre ich štruktúru, čo znamená, že k absorpcii dochádza, keď sa frekvencia žiarenia zhoduje s vibračnou frekvenciou molekuly. V tomto prípade, keď je fotón pohltý študovanou hmotou, ide o absorpčnú infračervenú spektroskopiou. V prípade, že je fotón hmotou vyžiarený, jedná sa o emisnú infračervenú spektroskopiou. Vo väčšine prípadov je využívaná práve absorpčná spektroskopia.

Infračervená spektroskopia sa zaoberá infračervenou časťou elektromagnetického spektra, čo je svetlo s väčšou vlnovou dĺžkou a nižšou frekvenciou, než má viditeľné svetlo. V elektromagnetickom spektre sa nachádza medzi viditeľným a mikrovlnným žiarením [25]. Infračervené svetlo sa podľa vlnových dĺžok žiarenia delí na ďaleké (400 - 10 cm<sup>-1</sup> / 25 - 100 μm vlnovej dĺžky), stredné (4000 - 400 cm<sup>-1</sup> / 2.5 - 25 μm vlnovej dĺžky) a blízke (14000 - 4000 cm<sup>-1</sup> / 0.7 - 0.25 μm vlnovej dĺžky), na základe jeho vzťahu k viditeľnému spektru. Blízke infračervené svetlo má vyššiu energiu a dokáže vyvolať podtón alebo harmonické vibrácie. Stredné môže byť použité pri štúdiu základných vibrácií a spojených rotačno-vibračných štruktúr. Ďaleké infračervené svetlo, ktoré leží najbližšie k mikrovlnnému svetlu, má nízku energiu a môže byť použité pri rotačnej spektroskopii. Pre identifikáciu a určovanie chemickej štruktúry má najväčší význam stredné infračervené žiarenie [24].

Infračervenou spektroskopiou je možné merať vzorky vo všetkých troch skupenstvách – pevnom, kvapalnom aj plynnom.

Používa sa, keď sú cieľom záujmu priestorovo odlišné vlastnosti a efekty, ohraničené zmeny, vrstevnaté štruktúry vzorky alebo rozlíšenie materiálov v ťažko oddeliteľných zmesiach.

Meria svetlo absorbované alebo odrazené od vzorky a výstupom je infračervené spektrum, ktoré je grafickým zobrazením funkčnej závislosti energie. Získané spektrum je potom možné využiť na kvalitatívnu ale aj kvantitatívnu analýzu vzorky [26]. Väčšinou býva vyjadrené v percentách priepustnosti alebo absorpcie, pričom priepustnosť je pomer intenzity žiarenia, ktoré prešlo vzorkou k intenzite žiarenia vychádzajúceho zo zdroja a absorpcia je definovaná ako dekadický logaritmus obrátenej hodnoty priepustnosti.

Svoje uplatnenie našla v elektrotechnickom priemysle, mineralógií, farmácií, pri analýze historických materiálov, textílií, papierov, pigmentov ale tiež vo forenzej chémii, kde sa pri analýze porovnáva namerané spektrum so spektrami v databáze známych spektier.

Na vykonávanie infračervenej spektroskopie sa používa infračervený spektrometer alebo spektrofotometer, ktorý produkuje infračervené spektrum [27].

#### **1.4.2 Ramanova spektroskopia**

Ramanova spektroskopia je nedeštruktívna technika využívaná na chemickú analýzu, ktorá poskytuje detailné informácie o chemickej štruktúre, fáze, polymorfii a vzájomnom pôsobení molekúl [31]. Je to technika založená na neelastickom rozptyle monochromatického svetla privádzaného z laseru s vysokou intenzitou, ktoré sa rozptýli pri kontakte s molekulou materiálu [4]. Väčšina odrazeného svetla má rovnakú vlnovú dĺžku alebo farbu ako mal jeho zdroj, čo sa nazýva Rayleighov rozptyl, ale malá časť rozptýleného svetla má inú vlnovú dĺžku alebo farbu, pretože sa pri kontakte s vibráciami molekúl zmenila jeho energia. Toto sa nazýva Ramanov rozptyl a jeho študovaním je možné zistiť chemické zloženie a iné užitočné informácie o vzorke [31].

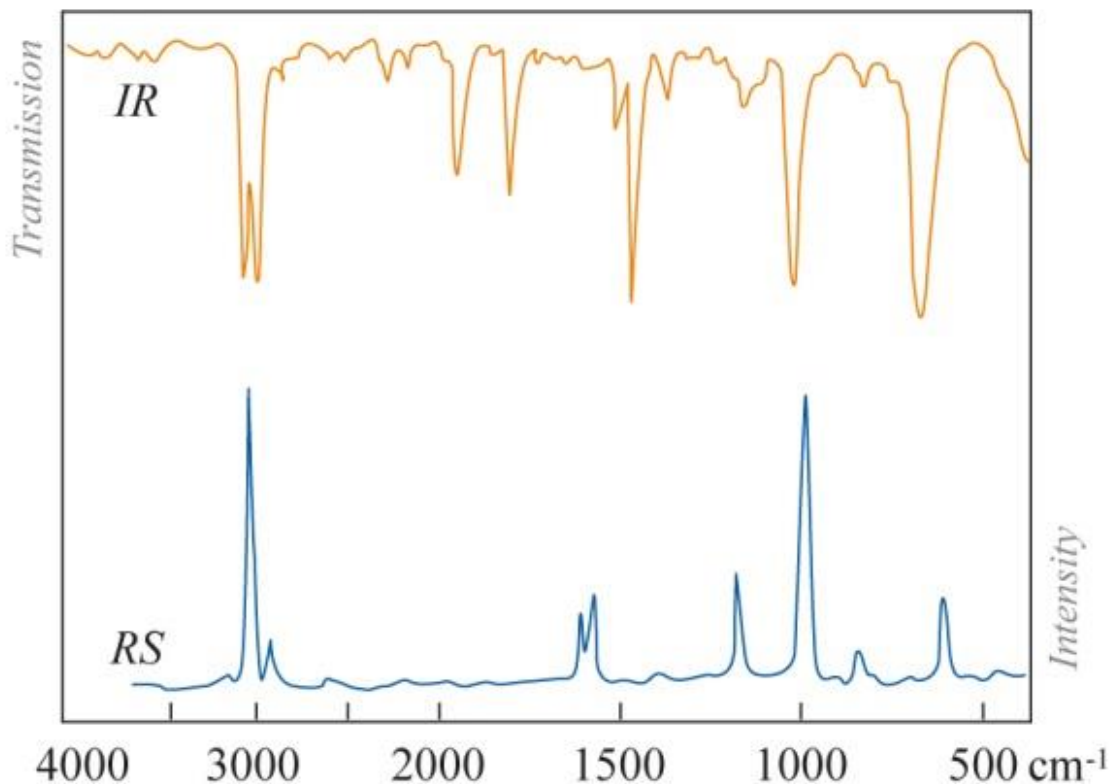
Počas tohto deja sa vymieňa energia medzi fotónmi a molekulou tak, že rozptýlený fotón má vyššiu alebo nižšiu energiu než pôvodný fotón. Rozdiel v energiách je spôsobený zmenami v rotačných a vibračných energiách molekuly [32].

„Princíp Ramanovej spektroskopie je teda v prechode svetelného lúča prostredím. Určitá časť je absorbovaná, časť prejde bez zmeny a posledná časť je rozptýlená. Svetelný lúč dopadá na molekuly látky a ak je zrážka dokonale pružná, svetlo sa rozptýli bez zmeny vlnovej dĺžky. Ak je zrážka nepružná dôjde k rozptylu – zmene smeru šírenia svetla – a

súčasne sa mení vlnová dĺžka svetla. Ak je dopadajúce svetlo monochromatické, bude v spektre rozptýleného žiarenia určitý počet čiar, ktorých vlnová dĺžka bude pri zrovnaní s pôvodnou posunutá.“ [34]

Ramanova spektroskopia je podobná infračervenej spektroskopii, ale zatiaľ čo infračervená spektroskopia skúma zmeny v dipólovom momente molekuly vyvolané interakciou so svetlom, Ramanova spektroskopia skúma zmeny polarizácie molekuly vyvolané tou istou interakciou. Keď sú energie z týchto prechodov zakreslené ako spektrum, dokážu identifikovať molekuly. Ramanove spektrá poskytujú doplnkovú informáciu k spektrám infračerveným. Vo väčšine prípadov sú zmeny intenzívne v Ramanových spektrách slabé v infračervených spektrách a naopak (Obrázok 16), pretože zmena polarizácie vibrácií sa zaznamená v Ramanovom spektre a zmena dipólu molekuly zase v infračervenom spektre. Preto budú intenzity vibračných frekvencií molekúl iné pri ich študovaní infračervenou a Ramanovou spektroskopiou [30].

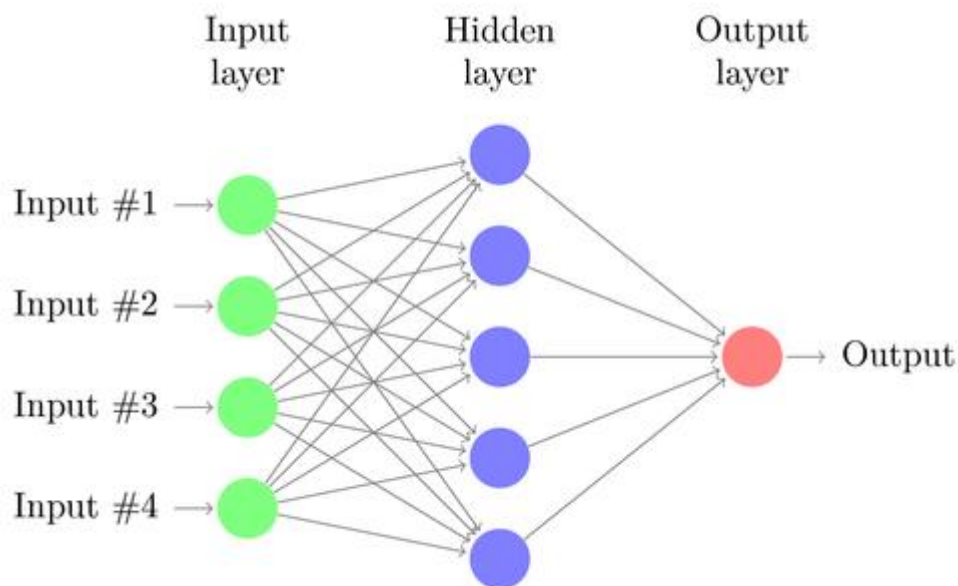
Ramanovu spektroskopiou je možné použiť na látky v tuhom, kvapalnom aj plynnom stave. Využíva sa pri určení štruktúry, kvalitatívnej analýze zmesí a vo forenznej chémii na analýzu vlákien, farieb, drog [29].



Obrázok 16 - Infračervená a Ramanova spektroskopia [33]

## 1.5 Neurónová sieť

„Softvérové neurónové siete patria do odboru umelej inteligencie, respektíve jej špecializovanej časti, označovanej ako deep learning (systémy hlbokého učenia sa).“ [21] Neurónová sieť je výpočtový model napodobňujúci biologické nervové systémy. Skladá sa z veľkého počtu prepojených procesných elementov – neurónov. Vo väčšine prípadov bývajú neurónové siete viacvrstvové, a teda zložené z viacerých vrstiev neurónov, pričom prvá vrstva sa nazýva vstupnou vrstvou, posledná vrstva výstupnou vrstvou a všetky vrstvy medzi nimi sú nazývané skryté vrstvy (Obrázok 17). Výstup každej vrstvy neurónov je zároveň vstupom nasledujúcej vrstvy a vstupom počiatkovej vrstvy sú priamo vstupné dáta. Čím viac vrstiev, tým zložitejšie dáta dokážu siete rozoznať, pretože agregujú a rekombinujú dáta z predchádzajúcej vrstvy. Toto umožňuje sieťam spracovávať veľké a mnohorozmerné údaje [19].

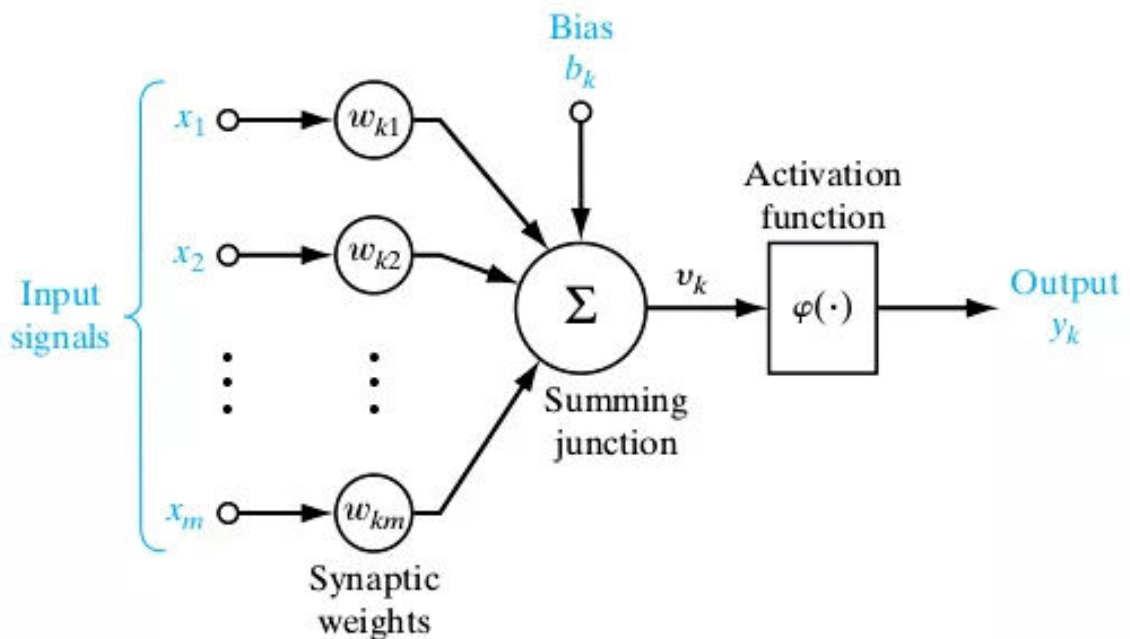


Obrázok 17 - Štruktúra neurónovej siete [23]

### 1.5.1 Neurón

Stavba neurónov je voľne založená na stavbe neurónov v ľudskom mozgu. V neurónových sieťach sú to miesta, kde sa vykonávajú výpočty a ktoré sa aktivujú pri určitých vstupoch, pričom vstupov môže byť viacero, ale výstup iba jeden. Neurón kombinuje vstup z dát so synaptickými váhami, ktoré tento vstup buď zosilnia alebo utlmia. Všetky takto vážené vstupy sú sčítané spolu s konštantným vstupom, ktorý môže nadobúdať hodnotu 1 alebo -1, váženým prahovou hodnotou (bias) a ich suma ďalej prechádza cez takzvanú aktivačnú

funkciu, aby sa určilo, či a do akej miery by mal signál postupovať ďalej cez sieť a ovplyvniť konečný výsledok (Obrázok 18). Ak signál prejde, neurón bol aktivovaný [20].



Obrázok 18 – Neurón [20]

### 1.5.2 Aktivačné funkcie

Aktivačná funkcia určuje výstup, ktorý neurón vygeneruje v závislosti na vstupe. Väčšinou býva:

- lineárna - kedy je výstupná aktivita priamoúmerná vstupu
- prahová - kedy má výstupná aktivita jednu z hodnôt  $[0, 1]$ , podľa toho, či je väčšia (výstup = 1) alebo menšia (výstup = 0) ako prahová hodnota
- sigmoidálna – kedy sa výstupná aktivita mení nelineárne podľa vstupu

### 1.5.3 Učenie

Základnou vlastnosťou neurónov a neurónových sietí je, že zapamätané poznatky získavajú prostredníctvom procesu učenia, pri ktorom sa upravujú synaptické váhy. Takýto neurón sa zmenou svojich váh alebo prahovej hodnoty dokáže adaptovať konkrétnej situácii. Učenie neurónových sietí môže byť dvojakého druhu – učenie s učiteľom, kedy neurónové siete klasifikujú vstupné dáta, pričom poznajú ich očakávaný výstup a učenie bez učiteľa, kedy zoskupujú neoznačené dáta na základe podobností medzi nimi [16], [5].



### **1.5.3.1 Učenie s učiteľom (klasifikácia)**

Pri učení s učiteľom existuje externý mechanizmus – učiteľ, ktorý má k dispozícii poznatky, ktoré sa má neurónová sieť naučiť. To znamená, že pre každú vstupnú vzorku zadá očakávaný výstup, aby sa neurónová sieť naučila vzájomné vzťahy medzi dátami a očakávanými výstupmi.

Neurón má dva režimy činnosti – trénovací režim a režim používania. V trénovacom režime, môže byť neurón natrénovaný, aby sa aktivoval (alebo nie) pri konkrétnych vstupoch. V režime používania, keď neurón zaznamená naučený vstup, naučený výstup je výstupom neurónu. Ak naučený vstup nie je zaznamenaný, je použitá aktivačná funkcia, aby rozhodla, či sa neurón aktivuje alebo nie.

Ak je sieť natrénovaná na trénovacích dátach, pričom niektoré neurón aktivujú a niektoré nie, potom dáta, ktoré neboli v týchto trénovacích dátach, aktivujú neurón vtedy, ak majú viac spoločných vlastností s dátami, ktoré spôsobia aktiváciu neurónu. Táto vlastnosť sa nazýva Hammingova vzdialenosť.

Dôležitou súčasťou učenia s učiteľom je problém konvergenencie chýb – minimalizácia chyby medzi očakávaným výstupom a sieťou vypočítaným výstupom. Cieľom je stanoviť váhy tak, aby minimalizovali túto chybu. Jednou zo známych využívaných metód je metóda najmenších štvorcov.

Učenie s učiteľom sa využíva napríklad pri rozpoznávaní tvárí, výrazov, gest, hlasov, identifikácii objektov, prekladu hovorenej reči na písmo a klasifikovaní emailov ako spam alebo podvod.

### **1.5.3.2 Učenie bez učiteľa (zhlukovanie)**

Pri učení bez učiteľa neexistuje externý mechanizmus – učiteľ, ktorý by tieto dáta klasifikoval. Toto učenie je založené na hľadaní podobností v trénovacích dátach. Čím viac dát, na ktorých môže algoritmus trénovať, tým presnejšie budú jeho výsledky. Je tiež nazývané samoorganizujúce sa, pretože samoorganizuje vstupné dáta a hľadá ich spoločné vlastnosti. Využívanými metódami sú Hebbovské učenie a kompetitívne učenie.

Učenie bez učiteľa sa využíva napríklad na porovnávanie dokumentov, obrázkov, zvukov a povrchov.

#### **1.5.4 Architektúra neurónových sietí**

Existuje niekoľko typov neurónových sietí. Ich architektúry sú implementované na základe matematických operácií a množiny parametrov potrebných na určenie výstupu a určujú vzájomné vzťahy medzi jednotlivými vrstvami neurónovej siete. Základnými architektúrami sú dopredné a rekurentné neurónové siete [16].

##### **1.5.4.1 Dopredné siete**

Dopredné neurónové siete nepripúšťajú, aby sa výstupné signály z neurónov vracali dozadu, dovoľujú šírenie signálov iba jedným smerom – od vstupu k výstupu. Neexistuje spätná väzba, takže výstup z vrstvy nikdy neovplyvní tú istú vrstvu. Dopredné neurónové siete sú priamymi sieťami, ktoré spájajú vstupy a výstupy. Používajú sa najviac pri rozpoznávaní vzorov.

##### **1.5.4.2 Rekurentné siete**

Rekurentné siete, alebo tiež siete so spätnou väzbou, pripúšťajú, aby sa výstupné signály z neurónov vracali dozadu. Majú teda signály, ktoré idú oboma smermi a výstup každého neurónu je privedený na vstup každého ďalšieho neurónu s výnimkou seba samého. Tieto siete sú dynamické, ich stav sa neustále mení, dokým nedosiahnu rovnovážny bod. Zostávajú v rovnovážnom bode, až kým sa nezmenia hodnoty vstupu a nie je potrebné vypočítať nový rovnovážny bod.

#### **1.5.5 Algoritmus spätného šírenia chyby**

Aby bolo možné trénovať neurónovú sieť, musíme prispôbiť váhy každého spojenia tak, aby chyba medzi želaným a reálnym výstupom bola čo najmenšia, pretože ak sa zmenia váhy, zmení sa aj chyba siete. Tento proces vyžaduje, aby neurónová sieť vypočítala chybu váh – ako sa mení chyba siete, keď sa menia váhy. Najpoužívanejšou metódou je algoritmus spätného šírenia chyby.

Algoritmus spätného šírenia chyby je ústredným mechanizmom, podľa ktorého sa neurónová sieť učí. Dopredné neurónové siete šíria signál dopredu až po výstupnú vrstvu. Tu je výstup siete porovnaný s očakávaným výstupom a je vypočítaná chyba siete ako ich rozdiel. Táto chyba sa potom šíri späť a použije sa na úpravu váh neurónovej siete. Proces sa opakuje, dokým nie je chyba minimálna.

Na zmenu váh sa využívajú gradientové algoritmy, ktoré hľadajú optimum funkcie – smer najstrmšieho stúpania alebo smer najstrmšieho spádu. Pri algoritme spätného šírenia chyby

sa využíva metóda hľadania smeru najstrmšieho spádu, ktorá hľadá minimum na minimalizáciu chyby siete [17].

#### **1.5.5.1 Gradient – výpočet zmeny hodnoty váh**

Gradient je jednou z najpoužívanejších optimalizačných funkcií chyby učenia na úpravu váh. Je to spád, ktorého uhol je možné zmerať a vyjadruje vzťah medzi dvomi premennými. V tomto prípade je to vzťah medzi chybou siete a každou z váh. Váhy majú podiel na chybe siete a ich zmenou sa môže táto chyba zväčšiť alebo zmenšiť. Výpočet zmeny hodnoty váh sa vypočíta ako podiel derivácie chyby siete k derivácii váhy. Pri algoritme spätného šírenia chyby sa potom váhy upravujú postupne - najprv váhy poslednej vrstvy, potom predposlednej až po prvú vrstvu.

### **1.6 Konvolučná neurónová sieť**

Konvolučné neurónové siete sú triedou dopredných neurónových sietí, ktoré sa primárne využívajú na klasifikáciu obrazov, ich zhukovanie podľa podobností a rozoznávanie objektov v nich. Využívajú skutočnosť, že na vstupe spracúvajú obrazovú informáciu, čo umožňuje zakódovať niektoré vlastnosti do ich architektúry a využiť ju tak rozumnejším spôsobom. Obrazy spracúvajú ako niekoľkorozmerné tenzory - matice čísel s dodatočnými dimenziami. Prvými dvomi rozmermi sú šírka a výška obrazu, tretím rozmerom je hĺbka – počet kanálov obrazu. Napríklad snímka s RGB kódovaním má hĺbku tri - jednu pre každú farbu v modeli. Konvolučná sieť vníma takúto snímku ako tri samostatné vrstvy farby naskladané jedna na druhej. Každá takáto vrstva sa nazýva kanál a počas konvolúcie vytvára mapy príznakov [18].

Konvolučná sieť je postupnosťou vrstiev a každá takáto vrstva transformuje vstupy na výstupy použitím rôznych funkcií. Na výstavbu konvolučných sietí sa využívajú najmä tri hlavné typy vrstiev, ktoré sú za sebou navrstvené pre vytvorenie kompletnej architektúry. Sú to:

- Konvolučná vrstva
- Subsamplingová (prevzorkovávacia) vrstva
- Plne prepojená vrstva (lineárna), ktorá má rovnakú architektúru ako klasická neurónová sieť

### **Konvolučná vrstva**

Konvolučná vrstva na vstupný obrázok aplikuje rôzne filtre, pričom každý reaguje na iný príznak. Môže to byť napríklad filter vyhľadávajúci horizontálne čiary, filter vyhľadávajúci vertikálne čiary alebo filter vyhľadávajúci šikmé čiary, ktoré vytvoria mapu hrán obrázku. Konvolučné siete aplikujú tieto filtre a vždy, keď je na obrázku nájdená zhoda, je namapovaná do mapy (tenzoru) zodpovedajúcej príznaku hľadanému daným filtrom. Takto je vytvorená mapa všetkých príznakov obrázku.

### **Prevzorkovávacia vrstva**

Ďalšou vrstvou v konvulčných neurónových sieťach, ktorá ďalej spracúva mapy príznakov (tenzory), je prevzorkovávacia vrstva. Prechádza cez tenzory pomocou malej  $n$ -rozmernej štvorcovej matice a do novej matice zapisuje najväčšiu hodnotu, ktorá sa nachádza v tenzore pod touto maticou. Zachovávajú sa teda iba príznaky s najväčšou koreláciou (najväčšou hodnotou) a všetky tieto hodnoty sú uložené do novej mapy  $n^2$ -krát menšej ako bola pôvodná. Ostatné príznaky sú vyradené.

Konvolučná a prevzorkovávacia vrstva sa môžu ľubovoľne veľa krát striedať, až je nakoniec výstup poslednej vrstvy privedený na vstup plne prepojenej lineárnej vrstvy neurónovej siete, ktorá funguje ako klasická neurónová sieť [22].

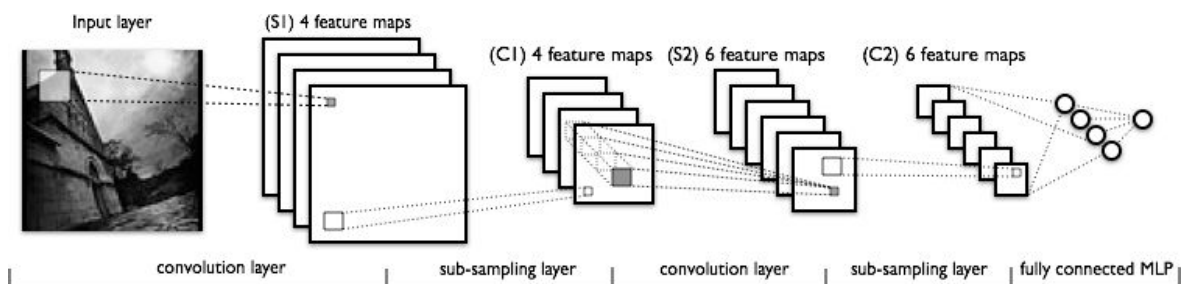
„Keď sú konvulčné neurónové siete trénované v rámci rozpoznávacieho procesu, vytvoria reprezentáciu obrázka, ktorého komplexnosť sa v rámci hierarchického procesu stupňuje.

Dá sa to mierne prirovnať k postupnému zaostrovaniu. Kým prvá vrstva určuje, kde sú na fotke svetlé a tmavé miesta, nasledujúca vrstva už začína rozpoznávať hrany, ďalšia tvary, nasledujúca objekty (napr. že na obrázku je človek), ďalšia typy objektov (muž) a napokon môže dôjsť aj k celkovému hodnoteniu situácie a pochopeniu významu (poloha muža, prítomnosť trávy a lopty indikuje, že hrá futbal).“ [21]

Na Obrázku 19 je možné vidieť fungovanie konvulčnej neurónovej siete za striedania dvoch konvulčných a dvoch prevzorkovávacích vrstiev ukončených plne prepojenou lineárnou vrstvou neurónovej siete [18].

1. Na vstupnom obrázku sú filtrami hľadané príznaky, pričom svetlý štvorec znázorňuje filter prechádzajúci cez obrázok. Výstupom sú mapy príznakov (tenzory), každá pre jeden filter.

2. Na vstupných tenzoroch sú hľadané najväčšie hodnoty, pričom biely štvorec predstavuje  $n$ -rozmernú štvorcovú maticu prechádzajúcu cez mapu. Tieto hodnoty sú potom zapisované do  $n^2$ -násobne menších nových máp príznakov.
3. V takto vytvorených tenzoroch sú opäť filtrom hľadané príznaky a sú zapisované do nových máp príznakov. Pre každý filter je vytvorená samostatná mapa. Tmavý štvorec znázorňuje filter prechádzajúci cez mapu.
4. Z nových tenzorov sú opäť vyberané najväčšie hodnoty, ktoré sú uložené do nových  $n^2$ -násobne menších máp. Biely štvorec predstavuje  $n$ -rozmernú štvorcovú maticu prechádzajúcu cez mapu.
5. Nakoniec sú tieto tenzory klasifikované pomocou plne prepojenej lineárnej neurónovej siete.



Obrázok 19 - Fungovanie konvolučnej neurónovej siete [18]

## **2 Návrh a implementácia metódy na identifikáciu vlákien vo forenznom skúmaní**

V praktickej časti sme sa zaoberali tvorbou softvéru na klasifikáciu mikroskopického obrazu textilného vlákna, ktorý sme overovali na reálnych dátach. Vstupom boli fotografie vlákien patriacich do pätnástich rôznych tried pod mikroskopom, tie boli rozdelené na menšie výrezy, ktoré boli následne upravené a rozdelené na trénovacie a testovacie dáta pre konvulučnú neurónovú sieť. Výstupom bol model neurónovej siete schopný klasifikovať dané vlákno do jednej z tried. Tento softvér bol napísaný v programovacom jazyku Python.

### **2.1 Príprava dát**

Pri príprave dát boli použité reálne vlákna zapožičané z katedry chémie, ktoré boli fotené pod mikroskopom pri desaťnásobnom priblížení. Z prírodných vlákien to boli bavlna, ríľovina, srst' mačky, vlna ovce, vlákna z velúru, srst' zajaca, juta a hodváb. Z umelých vlákien celofán, etylenvinylacetát (EVA), nylón, polyakrylnitril (PAN), polyetylén (PE), polyetyléntereftalát (PET) a polypropylén (PP). Vytvorených bolo približne 692 fotografií vlákien, pričom pre každú triedu zvlášť to bolo 58 fotografií (bavlna), 10 fotografií (ríľovina), 18 fotografií (srst' mačky), 25 fotografií (vlna ovce), 32 fotografií (vlákna z velúru), 78 fotografií (srst' zajaca), 12 fotografií (juta), 30 fotografií (hodváb), 11 fotografií (celofán), 11 fotografií (EVA), 141 fotografií (nylón), 25 fotografií (PAN), 34 fotografií (PE), 127 fotografií (PET) a 80 fotografií (PP).

### **2.2 Príprava dát pre neurónovú sieť**

Zhotovené fotografie ale zachytávajú veľkú časť vlákna, na ktorej sa často mení jeho zakrivenie a štruktúra, alebo je na nich zachytených viac vlákien, prípadne vzduchové bubliny alebo nečistoty. Takéto fotografie by boli na učenie sa pre neurónovú sieť nevhodné. Bolo preto potrebné vyrezať z týchto fotografií menšie časti zachytávajúce iba malú oblasť vlákna bez väčších zmien, so zhodným pomerom strán a rovnakou veľkosťou, ktorú sme zvolili 200 na 200 pixelov. Na pokrytie rôznych otočení vlákna a tiež na zväčšenie vstupnej vzorky sme tieto výrezy ďalej upravovali pomocou otáčania a nakoniec sme v každom výreze ešte zachytené vlákno vycentrovali. Úpravami pôvodných fotografií sa nám podarilo nedostačujúci počet dát značne rozšíriť a to na 63648 výrezov. Na ich prípravu sme používali funkcie z knižnice OpenCV, čo je Open Sourceová knižnica zameraná hlavne na počítačové videnie s rozhraním pre Python.

### 2.2.1 Vytvorenie menších výrezov vlákna

Na to, aby sme mohli z fotografie extrahovať výrezy vlákna, sme toto vlákno potrebovali najprv detegovať. Keďže väčšina fotografií zachytáva vlákno (Obrázok 20), poprípade vlákna, na jednofarebnom pozadí, rozhodli sme sa využiť metódu hľadania kontúr z knižnice OpenCV. Definície funkcií boli získané z oficiálnej dokumentácie tejto knižnice [8]. Definície ďalších použitých funkcií boli získané zo stránok oficiálnych dokumentácií ich knižníc – Scikit [10], Pillow [11] a SciPy [12].

#### Úprava obrázku pred hľadaním kontúr

Kvôli prípadom zachytávajúcim nečistoty, prípadne prerušené či splývajúce hrany alebo časti vlákna s pozadím, bolo potrebné fotografie pred samotným hľadaním kontúr ešte upraviť.

Na zväčšenie farebného rozdielu medzi vláknom a pozadím sme zvýšili kontrast:

```
enhancer = ImageEnhance.Contrast(image)
image = enhancer.enhance(6.0)
```

Na vyhladenie obrázku a odstránenie šumu sme použili mediánové vyhladenie (Obrázok 21), ktoré ako vstupné hodnoty vyžaduje:

- obrázok
- požadovanú veľkosť jadra

A výstupom je vyhladený obrázok. Mediánové vyhladenie nájde medián všetkých pixelov v jadre (ktoré má v našom prípade veľkosť 5 x 5 pixelov) a nahradí ním pixel v strede jadra.

```
image = cv2.medianBlur(image, 5)
```

Ďalej sme použili funkciu adaptívneho prahovania (Obrázok 22), ktorej výstupom je binárny obrázok a ktorá na vstupe vyžaduje:

- vstupný obrázok (src)
- maximálnu hodnotu (farbu), ktorá bude priradená tým pixelom, ktoré budú (binárne prahovanie) alebo nebudú (inverzné binárne prahovanie) spĺňať podmienku prahovania a to  $src(x,y) > T(x,y)$ ; kde  $T(x,y)$  je prahová hodnota vypočítaná pre každý pixel na základe hodnôt pixelov jeho okolia pomocou adaptívnej metódy prahovania

- adaptívnu metódu prahovania, kde máme na výber z dvoch metód - `ADAPTIVE_THRESH_MEAN_C`, ktorá  $T(x,y)$  pre daný bod počíta ako priemer bodov jeho okolia mínus  $C$  a `ADAPTIVE_THRESH_GAUSSIAN_C`, ktorá  $T(x,y)$  pre daný bod počíta ako váženú sumu jeho okolia mínus  $C$
- typ prahovania, kde máme na výber z dvoch metód - `THRESH_BINARY` (binárne prahovanie), ktoré pixelom pôvodného obrázku priraduje farby (čiernu alebo farbu vyjadrenú zadanou maximálnou hodnotou) podľa predpisu:

$$dst(x,y) = \begin{cases} \text{maximálna hodnota, } src(x,y) > T(x,y) \\ 0, & \text{inak} \end{cases} \quad (\text{Rovnica 1})$$

a `THRESH_BINARY_INV` (inverzné binárne prahovanie), ktoré pixelom pôvodného obrázku priraduje farby (čiernu alebo farbu vyjadrenú zadanou maximálnou hodnotou) podľa predpisu:

$$dst(x,y) = \begin{cases} 0, & \text{ak } src(x,y) > T(x,y) \\ \text{maximálna hodnota, } & \text{inak} \end{cases} \quad (\text{Rovnica 2})$$

- veľkosť okolia bodu
- konštantu  $C$

V našom prípade bol výstupom čierno-biely obrázok počítaný z okolia 11 x 11 pixelov za použitia inverzného binárneho prahovania a adaptívneho Gaussovho prahovania.

```
thresh = cv2.adaptiveThreshold(image, 255,
cv2.ADAPTIVE_THRESH_GAUSSIAN_C, cv2.THRESH_BINARY_INV, 11, 2)
```

Na odstránenie nežiaducich pixelov sme použili morfológickú operáciu erózie (Obrázok 23), ktorá v binárnom obrázku odstráni osamotené pixely a zmenší hrany. Na vstupe žiada:

- binárny vstupný obrázok
- maticu jadra

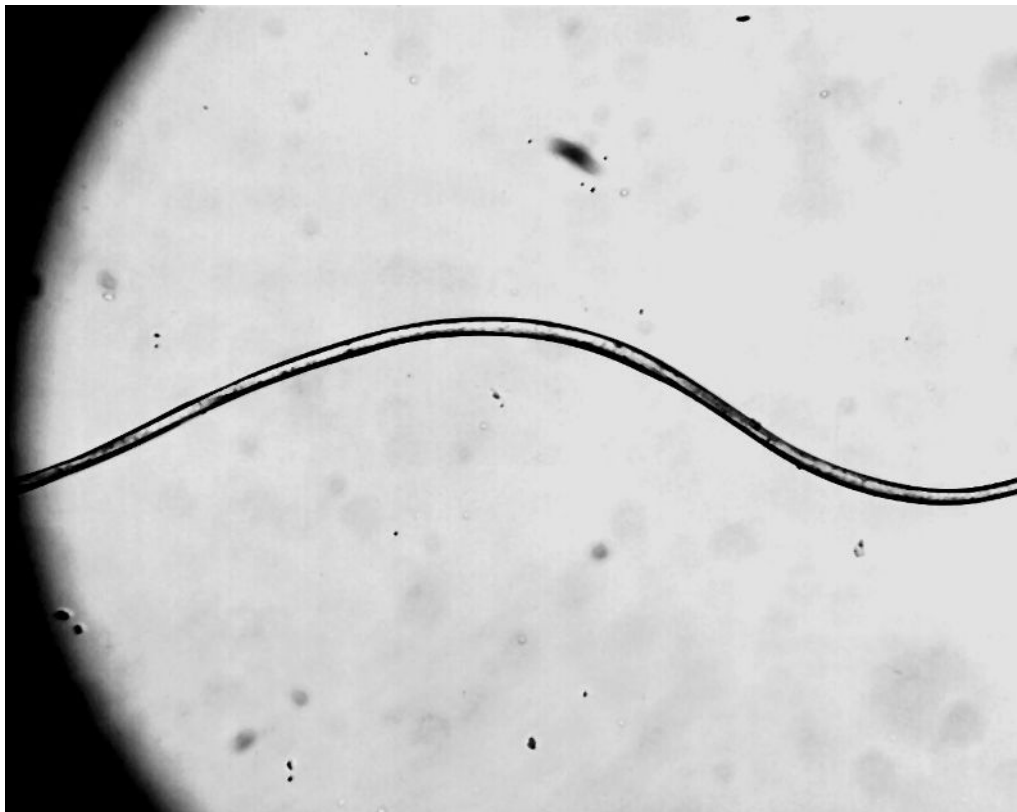
Matica jadra sa posúva po pôvodnom obrázku a výstupom je binárny obrázok, na ktorom sa nachádzajú hodnoty 1 práve vtedy, ak sa na vstupnom obrázku nachádzali hodnoty 1 pod všetkými hodnotami 1 v matici jadra. V našom prípade sme zvolili jadro o veľkosti 3x3 pixely jednotkovej matice.

```
kernel = np.ones((3, 3), np.uint8)
morf = cv2.erode(thresh, kernel, iterations=1)
```

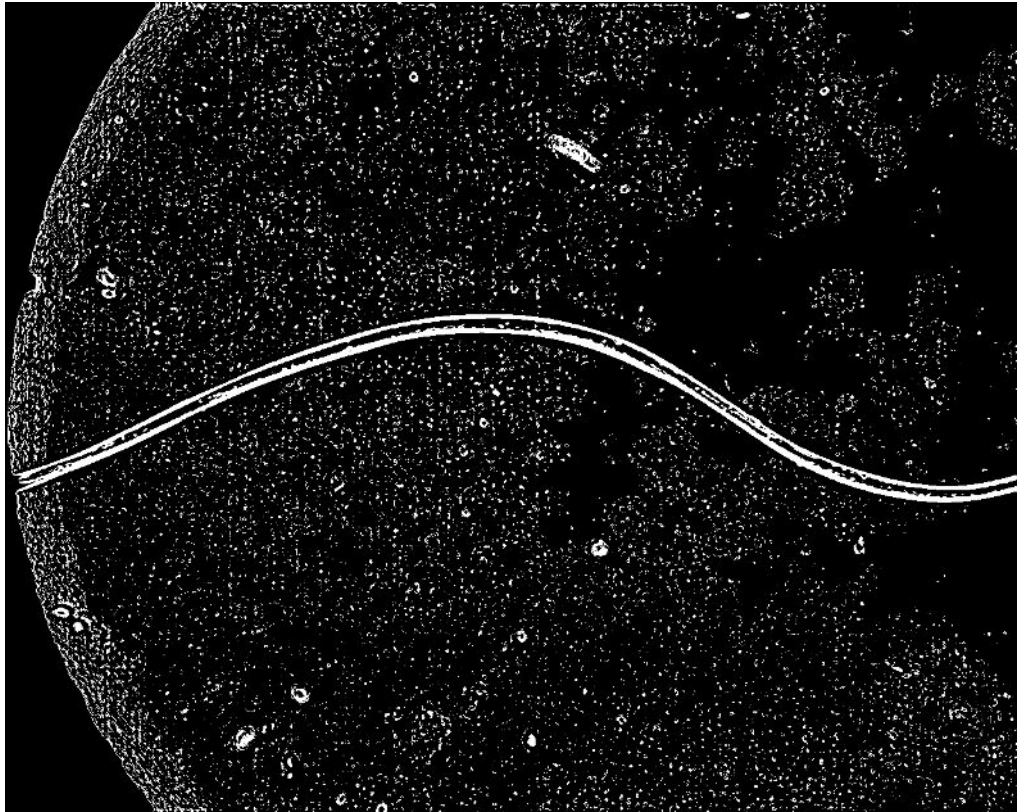




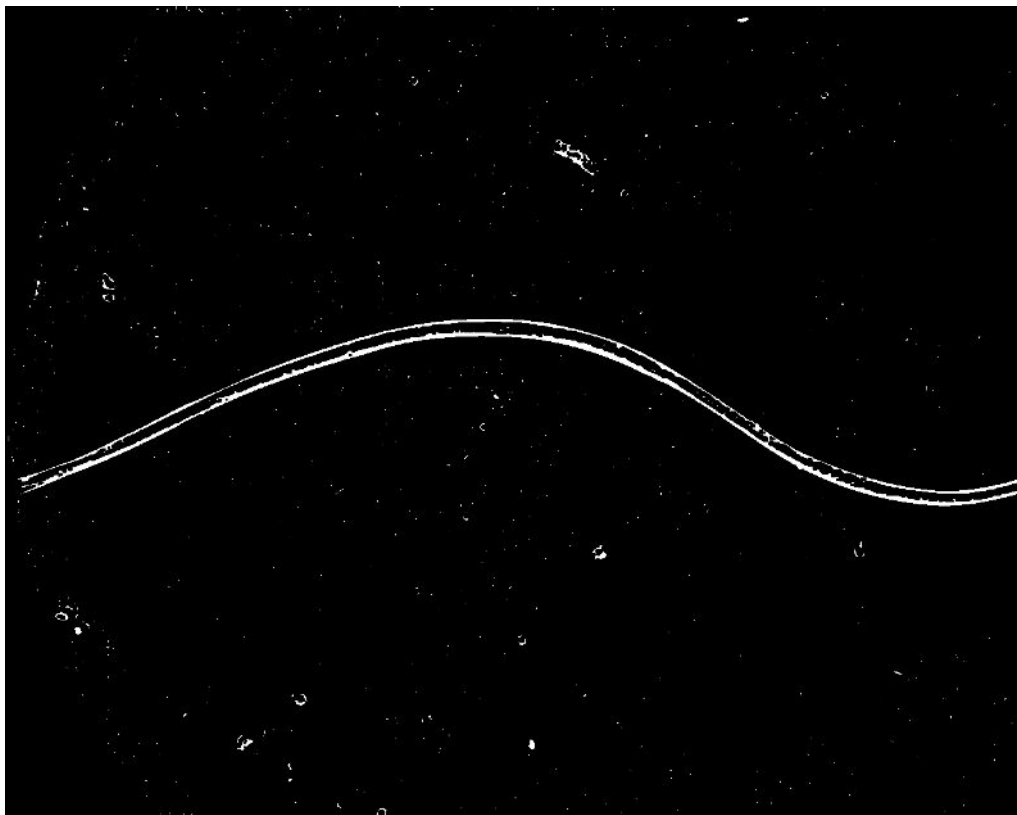
**Obrázok 20 - Pôvodná fotografia (zdroj: autor)**



**Obrázok 21 - Obrázok v škále šedej s uplatnenou funkciou mediánového rozmazania (zdroj: autor)**



Obrázok 22 - Binárny obrázok s uplatnenou funkciou adaptívneho prahovania (zdroj: autor)



Obrázok 23 - Binárny obrázok s uplatnenou morfológickou operáciou erózie (zdroj: autor)

## Hľadanie kontúr

Z takto upravených fotografií sme získavali kontúry vlákien (Obrázok 24). Na to sme použili funkciu knižnice OpenCV - findContours, ktorá na vstupe vyžaduje:

- vstupný obrázok
- režim vyhľadávania kontúr
- metódu aproximácie kontúr

Výstupom, ktorý nás zaujímal je zoznam kontúr detegovaných vo vstupnom obrázku, pričom každá kontúra je opísaná počtom súradníc x,y začiatočného a koncového bodu.

```
kontury, _ = cv2.findContours(morf, cv2.RETR_TREE,  
                             cv2.CHAIN_APPROX_SIMPLE)
```

Nájdené kontúry boli ďalej zotriedené zostupne podľa veľkosti plochy, ktorú zaberajú a tie, ktorých plocha bola väčšia ako zvolená prahová hodnota, boli spojené. Ich následnou aproximáciou pomocou funkcie approxPolyDP z knižnice OpenCV sme boli schopní opísať tvar kontúr menším množstvom úsečiek (Obrázok 25). Táto funkcia na vstupe vyžaduje:

- vstupný obrázok
- epsilon, ktorý predstavuje maximálnu vzdialenosť od kontúry k jej aproximácii a vypočíta sa ako konštanta \* dĺžka oblúka
- booleanové vyjadrenie, či je krivka uzavretá alebo nie

```
epsilon = 0.003 * cv2.arcLength(contours, True)  
approx = cv2.approxPolyDP(contours, epsilon, True)
```

Výsledkom aproximácie je pole súradníc x,y koncových a začiatočných bodov jednotlivých aproximovaných kontúr.

Tieto koncové body sme ďalej spájali do zhlukov pomocou metódy fit\_predict z knižnice Scikit, ktorá vykoná zhlukovanie pomocou zvolenej zhlukovacej metódy na daných dátach a pre každý bod v nich vráti index zhluku, ku ktorému prislúcha. Pre nás najvhodnejšie výsledky dávala zhlukovacia metóda MeanShift, ktorá patrí ku zhlukovacím metódam založeným na báze hustoty. Body k zhlukom priraduje iteratívne a to vždy k najbližšiemu centru zhluku, ktoré je v každom kroku vypočítané ako priemer všetkých bodov priradených k danému zhluk. Počet zhlukov nevyžaduje poznať dopredu, ale sama ho určí

podľa vstupných dát. Algoritmus MeanShift si nevyžaduje žiadne vstupné parametre, ale my sme sa mu rozhodli zadať ako vstupný parameter šírku jadra zhluky. V prípade, že vstupné parametre nemá zadané, dopočíta si ich tento algoritmus sám.

```
clustering = MeanShift(bandwidth=0.3)  
clusters = clustering.fit_predict(data)
```



**Obrázok 24 - Nájdené kontúry (zdroj: autor)**



**Obrázok 25 - Aproximácia kontúr (zdroj: autor)**



**Obrázok 26 - Centrá zhlukov bodov aproximácie (zdroj: autor)**



Obrázok 27 – Vlákna (v tomto prípade 3) nájdené programom rozlíšené farbami (zdroj: autor)

### Vyrezávanie vlákna

Na základe indexov vrátených funkciou `fit_predict` boli všetky body uložené do poľa prislúchajúceho zhluku v zozname zhlukov a pre každý zhluk bolo vypočítané jeho centrum ako priemer všetkých bodov v ňom (Obrázok 26). Ďalej bol vytvorený zoznam vlákien, pričom každé vlákno bolo reprezentované zoznamom zaradom usporiadaných centier zhlukov, ktoré ho tvoria. Centrum, v ktorého zhluku sa nachádzal najkrajnejší bod – prvý bod vo výslednom poli aproximácie, bolo zvolené za začiatkové centrum a bolo uložené do zoznamu kontúr prvého vlákna. K tomuto centru bola vypočítaná euklidovská vzdialenosť všetkých ďalších centier. Jemu najbližšie centrum, v ktorého zhluku sa nachádzal aspoň jeden bod ležiaci na spoločnej úsečke s aspoň jedným bodom zo zhluku prvého centra, bolo pridané do zoznamu kontúr ako ďalšie. Tento postup sa opakoval dokým existovalo ďalšie najbližšie centrum, ktorého zhluk mal so zhlukom predchádzajúceho centra aspoň jednu spoločnú kontúru. V prípade, ak nebolo možné nájsť ďalšie takéto centrum zhluku, ale ešte stále existovali nezaradené centrá zhlukov, hľadal algoritmus pre každé z nich najbližšie ešte nezaradené centrum zhluku, s ktorým malo spoločnú kontúru. Ak našiel dve centrá, ktoré si vzájomne vyšli ako najbližšie (a ich zhluky mali spoločnú kontúru), spojil ich do osobitného zoznamu,

vypočítal ich stredný bod a v ďalšom kroku hľadania najbližších nezaradených centier počítal s ním. Tento postup sa opakoval dokým existovalo ďalšie ešte nezaradené centrum, ktorého zhuk mal s niektorým ďalším zhukom ešte nezaradeného centra spoločnú kontúru. V prípade, že algoritmus nenašiel žiadne ďalšie takéto centrum zhuku, hľadal najbližšie už zaradené centrum zhuku so spoločnou kontúrou. Keď také našiel, priradil ho do zoznamu a celý tento zoznam uložil do zoznamu vlákien ako ďalšie vlákno (Obrázok 27).

Z takto vytvoreného zoznamu vlákien sme mohli začať vyberať výrezy. Pôvodne sme vytvárali nadväzujúce výrezy a posúvali sme sa po vlákne s veľkosťou kroku rovnou rádiusu (nami zvoleným na 200 pixelov), ale pretože pri neskorších úpravách dochádzalo pri takýchto výrezoch k strate obrazovej informácie, rozhodli sme sa zmenšiť krok na  $\text{rádius} / 3$ , pri ktorom vzájomné prekrytie zabezpečilo, že sa časti obrazu nestratili ani po úpravách.

Aby sme mohli počítať výrezy, bol vytvorený zoznam čiar medzi centrami zhukov. Ako prvý začiatkový bod bolo opäť zvolené centrum, v ktorého zhuku sa nachádzal najkrajnejší bod – prvý bod vo výslednom poli aproximácie. K tomuto bodu sa hľadal nasledujúci priesečník s prvou čiarou v zozname čiar vzdialený o rádius (Obrázok 28). V prípade, že sa priesečník nenašiel, bola táto čiara zo zoznamu vymazaná a postup sa opakoval s nasledujúcou. V prípade, že sa priesečník našiel, dopočítal program súradnice rohov štvorca, pričom začiatkový bod a priesečník boli brané ako stredy jeho protíahlych strán a nakoniec tento výrez uložil (Obrázok 29). Ako ďalší začiatkový bod potom zobral bod o  $\text{rádius} / 3$  vzdialený od predchádzajúceho začiatkového bodu a v cykle pokračoval, dokým zo zoznamu nevymazal poslednú čiaru. Na Obrázku 30 je možné vidieť výrezy s vyznačenými začiatkovými a koncovými bodmi, na Obrázku 31 všetky výrezy nájdené na danom vlákne a na Obrázku 32 jeden z uložených výrezov.

Tento postup ale nefungoval na 100%, pretože všetky kontúry na obrázku boli spojené do jednej kontúry a teda boli spojené aj body, medzi ktorými bol na obrázku prázdny priestor, poprípade program našiel kontúry vzduchovej bubliny, okraju sklíčka alebo nečistoty v preparáte. Taktiež výrezy, ktoré zobrazovali koncové časti vlákna, alebo časti, na ktorých bolo vlákno poškodené, sme považovali za nevhodné. Preto bolo ešte potrebné manuálne vymazať zlé výrezy.

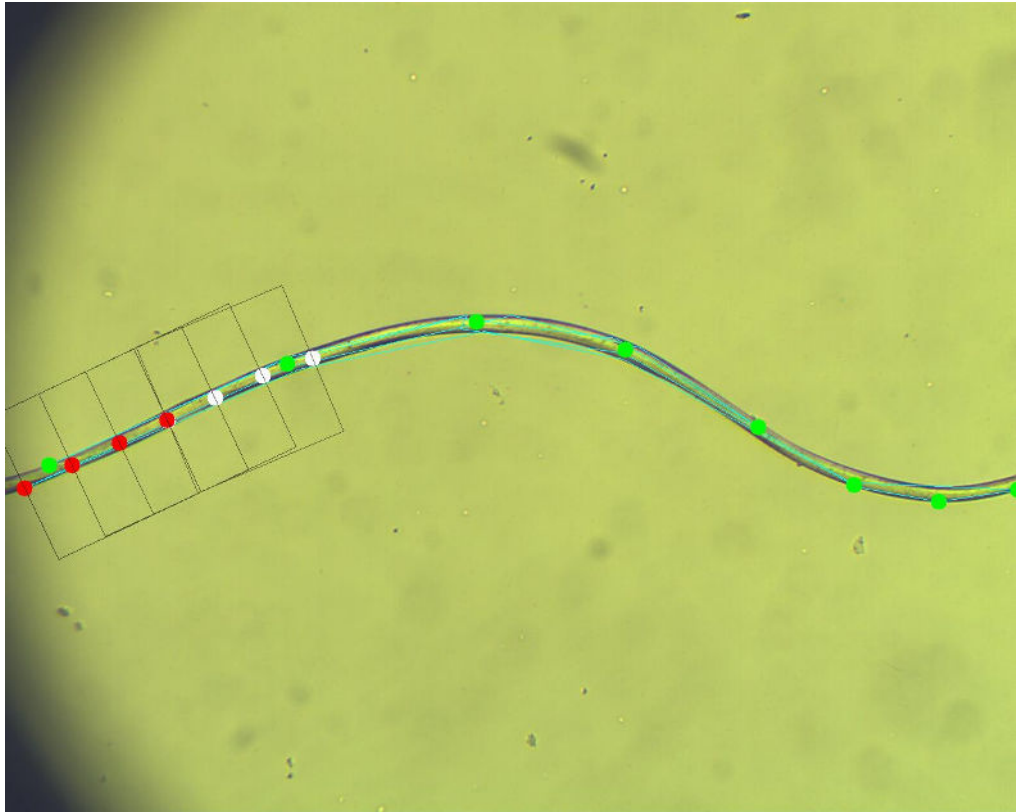


**Obrázok 28 - Začiatočný (červený) a koncový (biely) bod prvého výrezu (zdroj: autor)**



**Obrázok 29 - Začiatočný (červený) a koncový (biely) bod prvého výrezu s vyznačeným výrezom (zdroj: autor)**





**Obrázok 30 - Výrezy s vyznačenými začiatočnými (červenými) a koncovými (bielymi) bodmi (zdroj: autor)**



**Obrázok 31 - Výrezy vyznačené na pôvodnej fotografii (zdroj: autor)**



Obrázok 32 - Výrez vlákna

### 2.2.2 Úprava výrezov

Aby sme pokryli možnosti s rôznym sklonom vlákna a tiež poskytli neurónovej sieti väčšiu množinu dát na učenie, rozhodli sme sa získané výrezy ďalej upraviť otáčaním a transponovaním.

#### Otočenie a transponovanie

Každý výrez bol prevrátený vertikálne a horizontálne použitím funkcie transpose z knižnice PIL, ktorá si ako vstupný parameter vyžaduje metódu transponovania. Na výber je jedna z metód PIL.Image:

- .FLIP\_LEFT\_RIGHT – horizontálne prevrátenie
- .FLIP\_TOP\_BOTTOM – vertikálne prevrátenie
- .ROTATE\_90 – otočenie o 90 stupňov
- .ROTATE\_180 – otočenie o 180 stupňov
- .ROTATE\_270 – otočenie o 270 stupňov
- .TRANSPOSE – transponovanie

My sme využili metódu horizontálneho a vertikálneho prevrátenia.

```
image.transpose(Image.FLIP_TOP_BOTTOM)
```

```
image.transpose(Image.FLIP_LEFT_RIGHT)
```

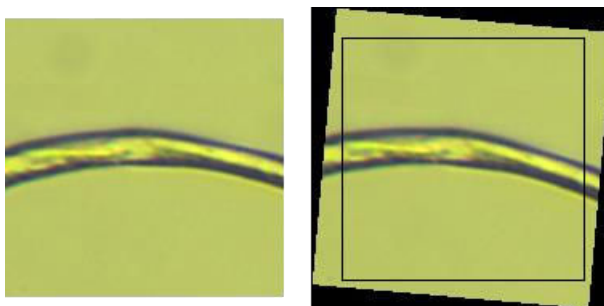
Každý pôvodný výrez bol ešte otočený o deväťdesiat stupňov a tiež postupne otáčaný o jednej stupeň v stanovenom rozmedzí -5 až 5 stupňov pomocou funkcie rotate z knižnice SciPy, ktorá si ako vstupné parametre vyžaduje:

- vstupný obrázok
- počet stupňov otočenia

```
rotate(image, 90)
```

Obe funkcie vracajú otočený obrázok.

Operácia otáčania po jednom stupni ale spôsobila deformáciu výrezu, ktorý bol na plnú veľkosť doplnený čiernymi okrajmi, ktoré sme potrebovali odstrániť. Potrebovali sme preto nájsť najväčší štvorec, ktorý by bolo možné vyrezať (Obrázok 33). Keďže najväčšiu deformáciu spôsobilo otočenie o 5 (respektíve -5) stupňov, vypočítali sme veľkosť a súradnice najväčšieho štvorca z takto otočeného výrezu a pretože neurónová sieť potrebuje na vstupe obrázky rovnakej veľkosti, vyrezali sme tento štvorec zo všetkých upravených výrezov a nové výrezy sme uložili. Práve tu dochádzalo k strate obrazovej informácie pri nadväzujúcich výrezoch, ktorá sa zamedzila ukladaním prekrývajúcich sa výrezov.



**Obrázok 33 - Otočenie obrázka - vľavo neotočený obrázok, vpravo otočený obrázok (zdroj: autor)**

### **Centrovanie**

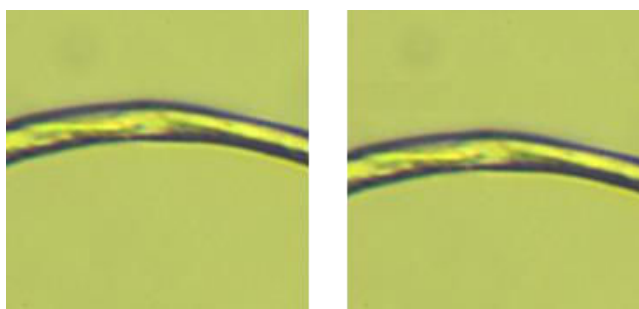
Ideálnym prípadom, ktorý sme potrebovali dosiahnuť, bolo, aby sa samotné vlákno nachádzalo v strede výrezu. To sa ale v niektorých prípadoch nestávalo a preto sme museli vlákno vo výreze nájsť a vycentrovať ho. Na detekciu vlákna vo výreze sme použili rovnaký postup ako na detekciu vlákna v pôvodnej fotografii a teda sme každému výrezu zvýšili kontrast, vyhladili sme ho mediánovým vyhladením, vytvorili sme z neho binárny obrázok pomocou adaptívneho Gaussovho prahovania, odstránili sme osamotené pixely a zmenšili hrany morfológickou operáciou erózie a nakoniec sme našli kontúry pomocou funkcie `findContours`. Kontúry sme zoradili zostupne podľa veľkosti, ale v tomto prípade sme ich už ďalej neaproximovali, ale rovno sme tie, ktorých veľkosť bola väčšia ako zvolená prahová hodnota, spojili do jednej kontúry. K tejto kontúre sme použitím funkcie `boundingRect` z knižnice `OpenCV` našli obdĺžnik, ktorý ju ohraničoval. Vstupným parametrom tejto funkcie je:

- pole bodov tvoriacich kontúry

A vracia x-ovú a y-ovú súradnicu ľavého horného rohu najmenšieho možného obdĺžnika, ktorý je vonkajším ohraničením týchto bodov a tiež jeho šírku a výšku.

```
x, y, w, h = cv2.boundingRect(contours)
```

V prípade, že tento obdĺžnik nezaberal celú veľkosť výrezu, bol vystrihnutý a vycentrovaný horizontálne – ak bola jeho šírka menšia ako jeho výška alebo vertikálne – ak bola jeho výška menšia ako jeho šírka. Takto vzniknutá chýbajúca časť bola nahradená časťou, ktorú tento výrez prekryl (Obrázok 34).



**Obrázok 34 - Centrovanie vlákna - vľavo nevycentrované vlákno, vpravo vycentrované vlákno (zdroj: autor)**

## 2.3 Neurónová sieť

Takto vytvorené výrezy sme použili ako vstupné dáta pre neurónovú sieť. Každú triedu sme rozdelili v pomere 80:20 na tréningové dáta, podľa ktorých neurónová sieť vytvorila model na klasifikáciu dát a na testovacie dáta, na ktorých sa overila správnosť tohto modelu. Na vytvorenie modelu neurónovej siete sme využívali knižnicu PyTorch., ktorá je Open Sourceovou deep learningovou knižnicou pre Python založenou na knižnici Torch. Definície funkcií boli získané z oficiálnej dokumentácie tejto knižnice [7].

Neurónovú sieť sme trénovali na 100 epochách.

### 2.3.1 Definovanie konvolučnej neurónovej siete

Konvolučná neurónová sieť sa skladá z niekoľkých na seba nadväzujúcich vrstiev. My sme v našom programe použili konvolučnú vrstvu, prevzorkovaciu vrstvu, lineárnu vrstvu, nulujúcu vrstvu (kde je výstup každého neurónu, ktorý je náhodne vybraný na vypadnutie, nastavený na nulu, bez ohľadu na jeho vstup), výplňovú vrstvu (ktorá vyplní okraje tenzora) a funkciu nelineárnej aktivácie, ktorá prahuje zadané hodnoty a môže byť použitá ako vrstva.

#### Konvolučná vrstva

Na definovanie konvolučnej vrstvy sme použili funkciu `torch.nn.Conv2d`, ktorá prechádza filtrom danej veľkosti po vstupných tenzoroch a hľadá s filtrom zhodné príznaky, ktoré zapisuje do nových tenzorov a ako parametre si vyžaduje:

- počet kanálov vstupu
- počet kanálov, ktoré má konvolučná vrstva vytvoriť
- veľkosť filtra prechádzajúceho po vstupe
- krok konvolúcie (veľkosť kroku, akým sa filter posúva po vstupe); tento parameter nie je povinný a v prípade, že nie je definovaný užívateľom, je jeho predvolená hodnota 1
- veľkosť nulami vyplnených okrajov pridaných na obe strany vstupu; tento parameter nie je povinný a v prípade, že nie je definovaný, je jeho predvolená hodnota 0
- rozostup medzi elementami filtra (1 = do filtra zoberie každý prvý element, 2 = do filtra zoberie každý druhý element, pričom počet elementov je vždy rovný veľkosti

filtra); tento parameter nie je povinný a v prípade, že nie je definovaný, je jeho predvolená hodnota 1

- počet spojení medzi vstupnými kanálmi a výstupnými kanálmi (1 = zo všetkých vstupov sú vytvorené výstupy, 2 = výstupy sú vytvorené akoby dvomi konvolučnými vrstvami, pričom každá z nich vidí polovicu vstupov a vytvorí polovicu výstupov a obe sa následne spoja); tento parameter nie je povinný a v prípade, že nie je definovaný, je jeho predvolená hodnota 1
- pridanie prahovej hodnoty - ak je zvolená hodnota True, pridá do siete učiacu sa prahovú hodnotu; tento parameter nie je povinný a v prípade, že nie je definovaný, je jeho predvolená hodnota True

```
self.conv = nn.Conv2d(3, 8, 7)
```

V tomto prípade konvolučná vrstva pracuje s trojkanálovým vstupom, pričom na výstupe z neho vytvorí osem kanálov (použije osem rôznych filtrov) a veľkosť matice filtra, ktorou sa bude posúvať po jednotlivých kanáloch vstupu bude 7x7.

Pri vstupnom tenzore v tvare  $(C_{in}, H_{in}, W_{in})$ , kde  $C_{in}$  je vstupný počet kanálov,  $H_{in}$  je vstupná výška a  $W_{in}$  je vstupná šírka tenzora, je výstupný tenzor definovaný ako  $(C_{out}, H_{out}, W_{out})$ , kde  $C_{out}$  je výstupný počet kanálov,  $H_{out}$  je výstupná výška a  $W_{out}$  je výstupná šírka tenzora. Pričom  $H_{out}$  sa vypočíta ako:

$$H_{out} = \left\lfloor \frac{H_{in} + 2 * \text{výplň}[0] - \text{rozostup}[0] * (\text{veľkosť filtra}[0] - 1) - 1}{\text{krok}[0]} + 1 \right\rfloor$$

(Rovnica 3)

a  $W_{out}$  sa vypočíta ako:

$$W_{out} = \left\lfloor \frac{W_{in} + 2 * \text{výplň}[1] - \text{rozostup}[1] * (\text{veľkosť filtra}[1] - 1) - 1}{\text{krok}[1]} + 1 \right\rfloor$$

(Rovnica 4)

### Prevzorkovávacia vrstva

Na definovanie prevzorkovávacej vrstvy sme použili funkciu `nn.MaxPool2d`, ktorá prechádza po vstupných tenzoroch maticou danej veľkosti a pre každú plochu na vstupnom

tenzore pod touto maticou vyberie maximálnu hodnotu, ktorú zapíše do nového tenzoru. Ako parametre si vyžaduje:

- veľkosť matice prechádzajúcej po vstupe, z ktorej je vyberaný maximálny prvok
- krok prevzorkovania (veľkosť kroku, akým sa matica posúva po vstupe)
- veľkosť nulami vyplnených okrajov pridaných na obe strany vstupu; tento parameter nie je povinný a v prípade, že nie je definovaný, je jeho predvolená hodnota 0
- rozostup medzi elementmi filtra (1 = do filtra zoberie každý prvý element, 2 = do filtra zoberie každý druhý element; pričom počet elementov je vždy rovný veľkosti filtra); tento parameter nie je povinný a v prípade, že nie je definovaný, je jeho predvolená hodnota 1
- vrátenie indexov - ak je jeho hodnota True, vráti maximálne indexy spolu s výstupmi); tento parameter nie je povinný a v prípade, že nie je definovaný, je jeho predvolená hodnota False
- mód zaokrúhlenia - ak je jeho hodnota True, zaokrúhli výstup nahor namiesto nadol); tento parameter nie je povinný a v prípade, že nie je definovaný, je jeho predvolená hodnota False

```
self.pool = nn.MaxPool2d(2, 2)
```

V tomto prípade bude veľkosť matice prevzorkovávacej vrstvy 2x2 a bude sa posúvať po krokoch veľkosti 2.

Pri vstupnom tenzore v tvare  $(C_{in}, H_{in}, W_{in})$ , kde  $C_{in}$  je vstupný počet kanálov,  $H_{in}$  je vstupná výška a  $W_{in}$  je vstupná šírka tenzora, je výstupný tenzor definovaný ako  $(C_{out}, H_{out}, W_{out})$ , kde  $C_{out}$  je výstupný počet kanálov,  $H_{out}$  je výstupná výška a  $W_{out}$  je výstupná šírka tenzora. Pričom  $H_{out}$  sa vypočíta ako:

$$H_{out} = \left\lfloor \frac{H_{in} + 2 * \text{výplň}[0] - \text{rozostup}[0] * (\text{veľkosť filtra}[0] - 1) - 1}{\text{krok}[0]} + 1 \right\rfloor$$

(Rovnica 5)

a  $W_{out}$  sa vypočíta ako:

$$W_{out} = \left\lfloor \frac{W_{in} + 2 * výplň[1] - rozostup[1] * (veľkosť filtra[1] - 1) - 1}{krok[1]} + 1 \right\rfloor$$

(Rovnica 6)

### Nulujúca vrstva

Na definovanie nulujúcej vrstvy sme použili funkciu `nn.Dropout`, ktorá počas tréovania náhodne nuluje niektoré elementy vo vstupných tenzoroch s pravdepodobnosťou  $p$  použitím vzoriek z Bernoulliho rozdelenia. Okrem toho sú počas tréovania výstupy škálované faktorom  $\frac{1}{1-p}$ . Ako parametre vyžaduje:

- pravdepodobnosť  $p$ , že bude element vynulovaný; tento parameter nie je povinný a v prípade, že nie je definovaný, je jeho predvolená hodnota 0.5
- zmena hodnoty v tenzore bez vytvorenia kópie - ak je jeho hodnota nastavená na `True`, zmení hodnotu elementu priamo v tenzore bez vytvorenia kópie; tento parameter nie je povinný a v prípade, že nie je definovaný, je jeho predvolená hodnota `False`

```
self.drop = nn.Dropout(0.5)
```

V tomto prípade bude nulujúca vrstva nulovať elementy s pravdepodobnosťou 0.5.

Výstupný tenzor má rovnaký tvar ako vstupný, takže pri vstupnom tenzore v tvare (C, H, W), kde C je počet kanálov, H je výška a W je šírka tenzora, je výstupný tenzor definovaný tiež ako (C, H, W).

### Výplňová vrstva

Na definovanie výplňovej vrstvy sme použili funkciu `nn.ConstantPad2d`, ktorá vyplní okraje vstupného tenzora konštantnou hodnotou a ako parametre vyžaduje:

- veľkosť okrajov
- konštantná hodnota, ktorou vyplní okraje

```
self.pad = nn.ConstantPad2d(3, 0)
```

V tomto prípade bude výplňová vrstva vyplňať okraje o veľkosti 3 po všetkých stranách vstupu a bude ich vyplňať hodnotou 0.



Pri vstupnom tenzore v tvare  $(C_{in}, H_{in}, W_{in})$ , kde  $C_{in}$  je vstupný počet kanálov,  $H_{in}$  je vstupná výška a  $W_{in}$  je vstupná šírka tenzora, je výstupný tenzor definovaný ako  $(C_{out}, H_{out}, W_{out})$ , kde  $C_{out}$  je výstupný počet kanálov,  $H_{out}$  je výstupná výška a  $W_{out}$  je výstupná šírka tenzora. Pričom  $H_{out}$  sa vypočíta ako:

$$H_{out} = H_{in} + \text{horný okraj} + \text{spodný okraj}$$

(Rovnica 7)

a  $W_{out}$  sa vypočíta ako:

$$W_{out} = W_{in} + \text{ľavý okraj} + \text{pravý okraj}$$

(Rovnica 8)

### Funkcia nelineárnej aktivácie

Na definovanie funkcie nelineárnej aktivácie sme použili funkciu `nn.PReLU`, ktorá prahuje zadané hodnoty podľa funkcie  $PReLU(x) = \max(0, x) + a * \min(0, x)$ . A teda ak je  $x$  väčšie ako 0, jeho hodnotu ponechá, v opačnom prípade ho prenásobí  $a$ , ktoré je učiacou sa váhou. Ako vstupné parametre si vyžaduje:

- počet  $a$  na učenie; zoberie iba dve hodnoty a to 1 alebo počet kanálov; tento parameter nie je povinný a v prípade, že nie je definovaný je jeho predvolená hodnota 1
- počiatočná hodnota  $a$ ; tento parameter nie je povinný a v prípade, že nie je definovaný, je jeho predvolená hodnota 0.25

```
self.prelu5 = nn.PReLU()
```

V tomto prípade bude nelineárna aktivácia prahovať zadané hodnoty s predvolenými parametrami (1, 0.25).

Výstupný tenzor má rovnaký tvar ako vstupný, takže pri vstupnom tenzore v tvare  $(C, H, W)$ , kde  $C$  je počet kanálov,  $H$  je výška a  $W$  je šírka tenzora, je výstupný tenzor definovaný tiež ako  $(C, H, W)$ .

### Lineárna vrstva

Na definovanie lineárnej vrstvy sme použili funkciu `nn.Linear`, ktorá aplikuje lineárnu transformáciu -  $y = x * A^T + b$  na vstupné tenzory a ako parametre vyžaduje:

- veľkosť každej vstupnej vzorky
- veľkosť každej výstupnej vzorky
- učenie sa prahovej hodnoty - ak je jeho hodnota nastavená na False, prahová hodnota vo vrstve sa nebude učiť; tento parameter nie je povinný a v prípade, že nie je definovaný, je jeho predvolená hodnota True

```
self.lin = nn.Linear(320, 19)
```

V tomto prípade bude veľkosť vstupnej vzorky pre lineárnu vrstvu 320 a veľkosť výstupnej vzorky bude 19.

### 2.3.2 Algoritmus neurónovej siete

Táto sieť bola vytvorená kombináciou a zret'azením konvolučnej, prevzorkovávacej, nulujúcej, výplňovej vrstvy a funkcie nelineárnej aktivácie a na ich výstup bola na záver použitá lineárna vrstva (Obrázok 35). Na Obrázku 36 je možné vidieť architektúru siete.

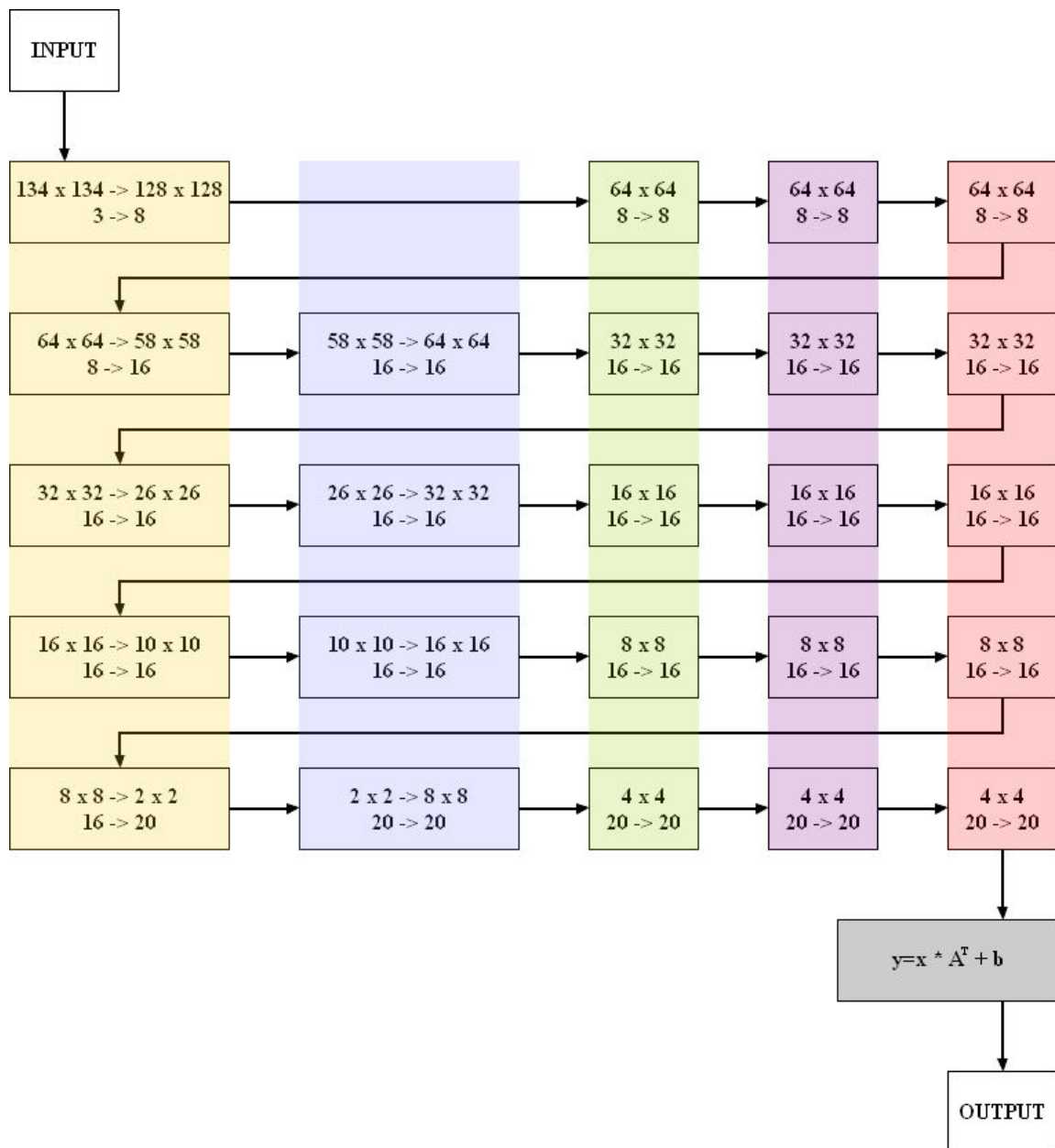
```
x = self.prelu1(self.drop(self.pool(self.conv1(x))))
x = self.prelu2(self.drop(self.pool(self.pad2(self.conv2(x)))))
x = self.prelu3(self.drop(self.pool(self.pad3(self.conv3(x)))))
x = self.prelu4(self.drop(self.pool(self.pad4(self.conv4(x)))))
x = self.prelu5(self.drop(self.pool(self.pad5(self.conv5(x)))))
x = self.fc1(x)
```

Obrázok 35 - Ukážka kódu modelu neurónovej siete

Tenzory trérovacej vzorky mali na začiatku veľkosť 134 x 134 x 3 (výška, šírka, počet kanálov). V každom kroku im bol konvolučnou vrstvou zväčšený počet výstupných kanálov (z troch postupne na 8, 16 až 20) a prevzorkovávacou vrstvou zmenšená veľkosť (podľa Rovnice 5 a 6) (postupne na 64, 32, 16, 8, 4).

Po prvom riadku sme teda dostali tenzor s veľkosťou 64 x 64 x 8, pretože funkcia *self.conv1()* je definovaná ako *self.conv1 = nn.Conv2d(3, 8, 7)* a teda berie tenzor s tromi vstupnými kanálmi, prechádza po ňom filtrom o veľkosti 7x7 a vytvorí tenzor s ôsmimi výstupnými kanálmi. Veľkosť strán výstupného tenzora sa pri tejto funkcii podľa Rovnice 3 a 4 zmení na 128 x 128 a bude pre 8 výstupných kanálov, preto bude výstupný tenzor veľkosti 128 x 128 x 8. Funkcia *self.pool()* je definovaná ako *self.pool = nn.MaxPool2d(2, 2)*, čo znamená, že bude s krokom veľkosti 2 prechádzať vstupný tenzor maticou 2x2 a z týchto 4 elementov pod maticou vyberie maximálny. Veľkosť strán výstupného tenzora teda bude v tomto prípade podľa Rovnice 5 a 6 dvojnásobne menšia pre výšku aj šírku

a teda 64x64. Počet kanálov sa pri tejto funkcii nemení, preto bude výstupný tenzor veľkosti 64 x 64 x 8. Funkcie *self.drop()* a *self.prelu()* veľkosť strán ani počet kanálov výstupného tenzora nijako neovplyvnia, preto bude v tvare 64 x 64 x 8 (výška, šírka, počet kanálov). Výsledné tenzory ďalších krokov vznikali obdobným spôsobom, s tým rozdielom, že vsunutá funkcia *self.pad()* definovaná ako *self.pad2 = nn.ConstantPad2d(3, 0)* vyplnila okraje o veľkosti 3 nulami, čím vrátila okraje stratené pri funkcii *self.conv()*. Na záver bola na výstupné tenzory veľkosti 4 x 4 x 20 aplikovaná funkcia *self.fc1()*, ktorá je definovaná ako *self.fc1 = nn.Linear(320, 19)* a predstavuje klasickú neurónovú sieť.



Obrázok 36 - Navrhnutá architektúra neurónovej siete (žltou - konvolučná vrstva, modrou – výplňová vrstva, zelenou – prevzorkovávacia vrstva, fialovou – nulujúca vrstva, červenou – funkcia nelineárnej aktivácie, šedou – lineárna vrstva) (zdroj: autor)

### **2.3.3 Módy neurónovej siete**

Neurónová sieť fungovala v dvoch módoch – v tréningovom, v ktorom sa z dát naučila charakteristiky jednotlivých tried a vytvorila model konvolučnej neurónovej siete a v testovacom, v ktorom použitím tohto modelu priradzovala dáta nepoužitú pri tréningu do daných tried. Dáta boli na tréningové a testovacie rozdelené v pomere 80:20, pričom výber do jednej alebo druhej skupiny bol náhodný, ale s náhodne zvoleným výrezom sa do danej skupiny presunuli aj všetky ostatné výrezy (a ich otočenia) z rovnakej fotografie.

#### **2.3.3.1 Tréningový mód**

V tréningovom móde sa neurónová sieť učila na 100 epochách z 51066 vzoriek, pričom pre každú epochu iterovala cez tréningovú množinu dát, a po vzorkách ich prechádzala algoritmom neurónovej siete a optimalizovala jej parametre (gradient). Zabralo jej to 183 minút a 35 sekúnd a jej priemerná chyba na 100 epochách bola 0.005659. Zatiaľ čo prvých 30 epoch sa chybovosť držala približne na 0.01, na zvyšných epochách už nevystúpila nad 0.005, pričom stále klesala, až nakoniec posledných 20 epoch oscillovala okolo 0.0017, čo bola výsledná chybovosť poslednej epochy. Výstupom tréningového módu bol model konvolučnej neurónovej siete.

#### **2.3.3.2 Testovací mód**

V testovacom móde overovala neurónová sieť vytvorený model na validačnej množine dát. Pri 12582 validačných obrázkoch jej to zabralo asi 8 sekúnd a jej úspešnosť bola 92%. Takto vysoká úspešnosť mohla byť čiastočne spôsobená obmedzenou množinou dát, z ktorej sme čerpali a tým, že viacero fotografií zachytávalo rôzne časti toho istého vlákna.

## Záver

Cieľom tejto práce bolo analyzovať možnosti klasifikácie mikroskopického obrazu textilného vlákna, navrhnúť a implementovať klasifikačný algoritmus a overiť ho na reálnych dátach.

V teoretickej časti sme sa venovali forenznej chémii a jej využitiu, ako aj niektorým druhom prírodných (bavlna, vlna) aj syntetických (nylon, polyester) vlákien. Na ich identifikáciu sa vo forenznej chémii využívajú rôzne metódy ako napríklad test dotykom, test horením, pozorovanie pod mikroskopom, chemické testy a spektroskopia, pri ktorej sa skúma vzájomné pôsobenie medzi hmotou a elektromagnetickým žiarením. Dvomi z najpoužívanejších metód spektroskopie vo forenznej vede sú infračervená spektroskopia, ktorá meria infračervené žiarenie absorbované alebo odrazené od vzorky a na výstupe poskytuje infračervené spektrum a Ramanova spektroskopia, ktorá študuje neelastický rozptyl monochromatického svetla, ktorému sa zmenila farba alebo vlnová dĺžka, keď sa pri kontakte s vibráciami molekúl zmenila jeho energia.

Pre náš prístup k identifikácii textilných vlákien sme sa rozhodli využiť fotografie vlákien pod mikroskopom a konvolučnú neurónovú sieť, ktorá sa využíva pri algoritmoch počítačového videnia na klasifikáciu obrazov, a rozpoznávanie objektov v nich. Je podskupinou dopredných neurónových sietí - výpočtových modelov napodobňujúcich biologické nervové systémy tvorených vrstvami (vstupnou, skrytými a výstupnou) neurónov - výpočtových jednotiek, ktoré si posielajú spracovávanú informáciu, ktorá sa v prípade dopredných sietí prenáša od vstupov k výstupom, a dokážu sa v procese tréningu neurónovej siete učiť.

Výstupom našej konvolučnej neurónovej siete bol model na klasifikáciu textilných vlákien, ktorý sa učil zo vstupných dát - štvorcových výrezov z fotografií vlákien fotených pod mikroskopom, rozdelených do jednotlivých tried. Tieto výrezy boli získané úpravou pôvodných fotografií, na ktorých boli nájdené kontúry vlákien, po ktorých sme prechádzali a vytvárali z nich výrezy, ktoré boli ďalej transformované, aby sa pokryli rôzne prípady otočenia vlákna vo výreze a tiež aby sa zväčšila vstupná vzorka. Následne boli vlákna v týchto výrezoch ešte vycentrované. Takto upravené dáta boli rozdelené na tréningovú a validačnú vzorku v pomere 80:20 a slúžili na tréning a testovanie neurónovej siete. Naša neurónová sieť bola zložená z niekoľkých zreťazených vrstiev – konvolučnej, ktorá na vstupných tenzoroch (maticiach čísel s dodatočnými dimenziami) hľadala príznaky;

prevzorkovávacej, ktorá zo vstupných tenzorov vyberala maximálne hodnoty pod zvolenou maticou; nulujúcej, ktorá náhodne nulovala niektoré elementy vo vstupných tenzoroch; výplňovej, ktorá vyplňala okraje tenzora konštantnou hodnotou; funkcie nelineárnej aktivácie, ktorá prahovala zadané hodnoty a lineárnej vrstvy, ktorá fungovala ako klasická neurónová sieť. V tréningovom móde sa naša neurónová sieť učila počas 100 epoch na množine 51066 tréningových dát, čo jej zabralo 183 minút a 35 sekúnd a jej priemerná chyba bola 0.005659. Jej výstupom bol model konvolučnej neurónovej siete, ktorého správnosť bola overená v testovacom móde na množine validačných dát a pri vzorke 12582 validačných obrázkov bola úspešnosť modelu 92%. Takto vysoká úspešnosť mohla byť čiastočne spôsobená obmedzenou množinou dát, z ktorej sme čerpali a tým, že viacero fotografií zachytávalo rôzne časti toho istého vlákna.

## Zoznam bibliografických odkazov

- [1] KHAN, E.A.N. et al. 2017. A Review Paper on Textile Fiber Identification. In *IOSR Journal of Polymer and Textile Engineering*. ISSN 2348-019X, 2017, vol. 4, no. 2, p. 14-20.
- [2] FERGUSSON, S.M. – HEMMING, J. 2018. *Forensic Examination of Fibres : Fibres, Yarns and Fabrics: An Introduction to Production, Structure and Properties*. 3. vyd. Boca Raton : Taylor & Francis Group. 2018. 486 s. ISBN 978-1-4398-2864-9.
- [3] FARAH, S. et al. 2015. Forensic comparison of synthetic fibers. In *Polymers for Advanced Technologies*. ISSN 1099-1581, 2015, vol. 26, no. 7, p. 785-796.
- [4] PRŮCHOVÁ, K. 2010. *Ramanova spektroskopie – tvorba a využití spektrální databáze* : bakalárska práca. Praha : Univerzita Karlova v Praze. 56 s.
- [5] HRINČÁR, M. 2012. *Konvoluční neuronové sítě a jejich využití při detekci objektů* : diplomová práca. Praha : Univerzita Karlova v Praze, 2012. 88s.
- [6] APPALANENI, K. 2013. *Non-Destructive Analysis of Trace Textile Fiber Evidence via Room-Temperature Fluorescence Spectroscopy* : dizertačná práca. Orlando : M.S. University of Central Florida, 2013. 169 s.
- [7] *PyTorch : Torch.nn*. [online]. [cit. 2019-04-16]. Dostupné na internete: <<https://pytorch.org/docs/stable/nn.html>>
- [8] *OpenCV : Image Processing*. [online]. [cit. 2019-04-16]. Dostupné na internete: <<https://docs.opencv.org/2.4/modules/imgproc/doc/imgproc.html>>
- [9] *Scikit : Clustering*. [online]. [cit. 2019-04-16]. Dostupné na internete: <<https://scikit-learn.org/stable/modules/clustering.html>>
- [10] *Scikit : sklearn.cluster.MeanShift*. [online]. [cit. 2019-04-16]. Dostupné na internete: <<https://scikit-learn.org/stable/modules/generated/sklearn.cluster.MeanShift.html>>
- [11] *Pillow*. [online]. [cit. 2019-04-16]. Dostupné na internete: <<https://pillow.readthedocs.io/en/3.1.x/reference/Image.html>>

- [12] SciPy. [online]. [cit. 2019-04-16]. Dostupné na internete: <<https://docs.scipy.org/doc/scipy/reference/generated/scipy.ndimage.rotate.html>>
- [13] *Textile Fashion Study*. [online]. [cit. 2019-04-16]. Dostupné na internete: <<http://textilefashionstudy.com/category/fibre/>>
- [14] Burn Test Chart. In *Fabric Mart*. [online]. [cit. 2019-04-16]. Dostupné na internete: <<https://www.fabricmartfabrics.com/Burn-Test-Chart.html>>
- [15] DRAGANI, R. 2018. Difference Between Nylon 6 & Nylon 66. In *Sciencing*. 2018. [online]. [cit. 2019-04-16]. Dostupné na internete: <<https://sciencing.com/difference-nylon-6-nylon-66-7617480.html>>
- [16] A Beginner's Guide to Neural Network. In *Skymind*. [online]. [cit. 2019-04-16]. Dostupné na internete: <<https://skymind.ai/wiki/neural-network>>
- [17] A Beginner's Guide to Backpropagation in Neural Networks. In *Skymind*. [online]. [cit. 2019-04-16]. Dostupné na internete: <<https://skymind.ai/wiki/backpropagation>>
- [18] A Beginner's Guide to Convolutional Neural Networks (CNNs). In *Skymind*. [online]. [cit. 2019-04-16]. Dostupné na internete: <<https://skymind.ai/wiki/convolutional-network>>
- [19] STERGIOU, CH. – SIGANOS, D. *Neural Network*. [online]. [cit. 2019-04-16]. Dostupné na internete: <[https://www.doc.ic.ac.uk/~nd/surprise\\_96/journal/vol4/cs11/report.html](https://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html)>
- [20] MANNARSWAMY, S. 2017. Everything you need to know about neural networks. In *OpenSourceForU*. 2017. [online]. [cit. 2019-04-16]. Dostupné na internete: <<https://opensourceforu.com/2017/03/neural-networks-in-detail/>>
- [21] URBAN, F. 2016. Človek vs. konvolučné neurónové siete. In *TouchIt*. 2016. [online]. [cit. 2019-04-16]. Dostupné na internete: <<https://touchit.sk/clovek-vs-konvolucne-neuronove-siete/58956>>
- [22] *CS231n Convolutional Neural Networks for Visual Recognition*. [online]. [cit. 2019-04-16]. Dostupné na internete: <<http://cs231n.github.io/convolutional-networks/>>



- [23] FAUSKE, K.M. 2006. Example: Neural network. In *TEXample*. 2006. [online]. [cit. 2019-04-16]. Dostupné na internete: <<http://www.texample.net/tikz/examples/neural-network/>>
- [24] *Infrared Spectroscopy*. [online]. [cit. 2019-04-16]. Dostupné na internete: <<https://www.ru.nl/systemschemistry/equipment/optical-spectroscopy/infrared/>>
- [25] REUSCH, W. *Infrared Spectroscopy*. [online]. [cit. 2019-04-16]. Dostupné na internete: <<https://www2.chemistry.msu.edu/faculty/reusch/virttxtjml/spectrpy/infrared/infrared.htm>>
- [26] MORAVEC, Z. Infračervená spektroskopie. In *Web o chemii, elektronice a programování*. [online]. [cit. 2019-04-16]. Dostupné na internete: <<http://z-moravec.net/chemie/fyzikalne-chemicke-metody/infracervena-spektroskopie/>>
- [27] SULOVSKEÝ, P. *Infračervená spektroskopie*. [online]. [cit. 2019-04-16]. Dostupné na internete: <[http://www.sci.muni.cz/~sulovsky/Vyuka/Lab\\_metody/Infracervena\\_spektroskopie.pdf](http://www.sci.muni.cz/~sulovsky/Vyuka/Lab_metody/Infracervena_spektroskopie.pdf)>
- [28] KANIA, P. *Infračervená spektrometrie*. [online]. [cit. 2019-04-16]. Dostupné na internete: <<https://www.vscht.cz/files/uzel/0005766/Infra%C4%8Derven%C3%A1+spektrometrie.pdf?redirected>>
- [29] *Introduction to Raman Spectroscopy*. [online]. [cit. 2019-04-16]. Dostupné na internete: <<http://bwtek.com/raman-introduction-to-raman-spectroscopy/>>
- [30] *Raman Spectroscopy – A Tutorial*. [online]. [cit. 2019-04-16]. Dostupné na internete: <[https://www.kosi.com/na\\_en/products/raman-spectroscopy/raman-technical-resources/raman-tutorial.php](https://www.kosi.com/na_en/products/raman-spectroscopy/raman-technical-resources/raman-tutorial.php)>
- [31] *What is Raman Spectroscopy?*. [online]. [cit. 2019-04-16]. Dostupné na internete: <[https://www.horiba.com/en\\_en/raman-imaging-and-spectroscopy/](https://www.horiba.com/en_en/raman-imaging-and-spectroscopy/)>
- [32] *A basic overview od Raman spectroscopy*. [cit. 2019-04-16]. Dostupné na internete: <<https://www.renishaw.com/en/a-basic-overview-of-raman-spectroscopy--25805>>

- [33] Vibrational spectroscopy. [online]. [cit. 2019-04-16]. Dostupné na internete: <<http://eng.thesaurus.rusnano.com/wiki/article1211>>
- [34] *Ramanova spektroskopia*. [online]. [cit. 2019-04-16]. Dostupné na internete: <<http://www.slovcert.sk/media/var/files/SciAps/Ramanova%20spektroskopia.pdf>>
- [35] BALL-DESLICH, B. 2009. Forensic Science for High School. 2. vyd. Dubuque : Kendall/Hunt Publishing, 2009. 548 s. ISBN 0757585523.

## **Prílohy**

### **Príloha A – Používateľská príručka**

**UNIVERZITA MATEJA BELA V BANSKEJ BYSTRICI**

**FAKULTA PRÍRODNÝCH VIED**

## **IDENTIFIKÁCIA TEXTILNÝCH VLÁKIEN VO FORENZNOM SKÚMANÍ**

### **Používateľská príručka**

**Banská Bystrica, 2019**

**Bc. Michaela Samuelčíková**

Táto príručka je napísaná pre prostredie PyCharm.

Pred spustením programu je potrebné mať fotografie vlákien, ktoré chceme použiť, uložené v priečinku v samostatných priečinkoch roztriedených podľa tried (Obrázok 37).



Obrázok 37 - Priečinky tried

Jednotlivé kroky sú ďalej rozdelené do štyroch programov:

1. find\_contours.py – hľadanie kontúr
2. transform.py – transformácia výrezov
3. center.py – centrovanie vlákien vo výrezoch
4. neural\_network.py – neurónová sieť

### Hľadanie kontúr

Ako prvý spustíme program na hľadanie kontúr a to otvorením programu find\_contours.py v prostredí PyCharmu, kde je možné zmeniť (Obrázok 38):

- radius – veľkosť štvorcového výrezu vyrezávaného z vlákien
- step – veľkosť kroku, ktorým sa tieto výrezy posúvajú po vlákne
- path – cesta k priečinku, v ktorom sú uložené priečinky roztriedené podľa tried vlákien (obsahujúce fotografie vlákien)
- final\_folder – cesta k priečinku, do ktorého chceme výsledné výrezy uložiť

a následným kliknutím pravým tlačidlom na názov programu (alebo do tela programu) a kliknutím na Run 'find\_contours' alebo kliknutím na zelenú šípku v pravom hornom rohu.

V závislosti od počtu fotografií vlákien môže tento program bežať niekoľko sekúnd, minút alebo hodín.

```

radius = 200
step = 3
path = ".\\fibers\\"
final_folder = ".\\fibers_cutted\\"

list_of_folders = os.listdir(path)
for folder in list_of_folders:
    list_of_fibers = os.listdir(path+folder)
    for fiber in list_of_fibers:
        im = Image.open(path + folder + "\\" + fiber)
        img = cv2.cvtColor(np.array(im), cv2.COLOR_RGB2BGR)

```

Obrázok 38 – Ukážka programu na hľadanie kontúr

### Transformácia výrezov

Po zbehnutí programu na hľadanie kontúr spustíme program na transformáciu výrezov a to otvorením programu transform.py v prostredí PyCharmu, kde je možné zmeniť (Obrázok 39):

- path – cesta k priečinku, v ktorom sú uložené priečinky roztriedené podľa tried vlákien (obsahujúce výrezy vlákien)
- final\_folder – cesta k priečinku, do ktorého chceme výsledný výrezy uložiť

a následným kliknutím pravým tlačidlom na názov programu v bočnom menu (alebo do tela programu) a kliknutím na Run ‘transform‘ alebo kliknutím na zelenú šípku v pravo hornom rohu.

V závislosti od počtu výrezov môže tento program bežať niekoľko sekúnd, minút alebo hodín.

```

path = ".\\fibers_cutted\\"
final_folder = ".\\fibers_transformed\\"

list_of_folders = os.listdir(path)
for folder in list_of_folders:
    list_of_fibers = os.listdir(path+folder)
    for fiber in list_of_fibers:
        degrees = -5
        im = Image.open(path+folder+"\\"+fiber)

```

Obrázok 39 – Ukážka programu na transformáciu výrezov

## Centrovanie vlákien vo výrezoch

Po zbehnutí programu na transformáciu výrezov spustíme program na centrovanie vlákien vo výrezoch a to otvorením programu center.py v prostredí PyCharmu, kde je možné meniť (Obrázok 40):

- path – cesta k priečinku, v ktorom sú uložené priečinky roztriedené podľa tried vlákien (obsahujúce transformované výrezy vlákien)
- final\_folder – cesta k priečinku, do ktorého chceme výsledný výrezy uložiť

a následným kliknutím pravým tlačidlom na názov programu v bočnom menu (alebo do tela programu) a kliknutím na Run 'center' alebo kliknutím na zelenú šípku v pravo hornom rohu.

V závislosti od počtu výrezov môže tento program bežať niekoľko sekúnd, minút alebo hodín.

```
path = ".\\fibers_transformed\\"
final_folder = ".\\fibers_centered\\"

list_of_folders = os.listdir(path)
for folder in list_of_folders:
    list_of_fibers = os.listdir(path+folder)
    for fiber in list_of_fibers:
        im = Image.open(path + folder + "\\" + fiber)
        img = cv2.cvtColor(np.array(im), cv2.COLOR_RGB2BGR)
```

Obrázok 40 – Ukážka programu na centrovanie vlákien vo výrezoch

## Neurónová sieť

Po zbehnutí programu na centrovanie vlákien vo výrezoch je potrebné výsledné výrezy rozdeliť na tréningovú a testovaciu vzorku (napríklad v pomere 80:20) tak, že v priečinku budeme mať dva priečinky s názvom 'train' a 'val' a v každom z nich budeme mať v samostatných priečinkoch roztriedené výsledné výrezy podľa tried. Potom spustíme program na tréningovanie neurónovej siete a to otvorením programu neural\_network.py v prostredí PyCharmu, kde je možné meniť (Obrázok 41):

- data\_dir – cesta k priečinku, v ktorom sú uložené priečinky 'train' a 'val' a v nich sú uložené priečinky roztriedené podľa tried vlákien (obsahujúce výsledné výrezy vlákien)

- `crop_size` – veľkosť vstupných výrezov
- `num_epochs` – počet epoch, počas ktorých sa bude program učiť
- `batch_size` – veľkosť vzorky
- `device` – názov zariadenia, na ktorom sa bude program spúšťať
- `model_name` – názov vytvoreného modelu

a následným spustením príkazu `./neural_network.py` (pre Linux) alebo `python neural_network.py` (pre Windows) v príkazovom riadku zariadenia, na ktorom má program bežať.

V závislosti od počtu výrezov môže tento program bežať niekoľko hodín až niekoľko dní.

Po zbehnutí programu na trénovanie neurónovej siete spustíme program na testovanie neurónovej siete, ktorý spustíme príkazom `./neural_network.py nazov_modelu` (pre Linux) alebo `python neural_network.py nazov_modelu` (pre Windows) v príkazovom riadku zariadenia, na ktorom má program bežať.

V závislosti od počtu výrezov môže tento program bežať niekoľko sekúnd až minút.

```

data_dir = "data-orig-div/"
crop_size = 134
num_epochs = 100
batch_size = 32
device = torch.device("cuda")
mean = [0.5, 0.5, 0.5]
std = [0.5, 0.5, 0.5]

mode = 'train'
model_name = 'fibers_model.pt'

```

Obrázok 41 – Ukážka programu neurónovej siete

**Príloha B – Systémová dokumentácia**

**UNIVERZITA MATEJA BELA V BANSKEJ BYSTRICI**

**FAKULTA PRÍRODNÝCH VIED**

**IDENTIFIKÁCIA TEXTILNÝCH VLÁKIEN VO FORENZNOM  
SKÚMANÍ**

**Systémová dokumentácia**

**Banská Bystrica, 2019**

**Bc. Michaela Samuelčíková**



Pre spustenie a úpravu programov je potrebné mať nainštalovaný jazyk Python (<https://www.python.org/>). Pre tvorbu programov sme použili nasledujúci softvér:

PyCharm – na inštaláciu sme stiahli inštaláčny súbor z oficiálnej stránky [jetbrains.com/pycharm/](https://jetbrains.com/pycharm/)

Na inštaláciu knižníc do PyCharmu je možné použiť dva spôsoby:

1. V programe PyCharm kliknúť na File ⇒ Settings (Obrázok 42) ⇒ Project ⇒ Project Interpreter (Obrázok 43), kde sa po kliknutí na + v pravom hornom rohu otvorí okno s dostupnými balíčkami (Obrázok 44) a po zvolení želaného balíčku kliknutím na `nazov_balicku` sa sprístupní možnosť Install Package v ľavom dolnom rohu; kliknutím sa balíček nainštaluje
2. V príkazovom riadku v priečinku s projektom spustiť príkaz `pip install nazov_balicku` (Obrázok 45)

Takto sme nainštalovali knižnice:

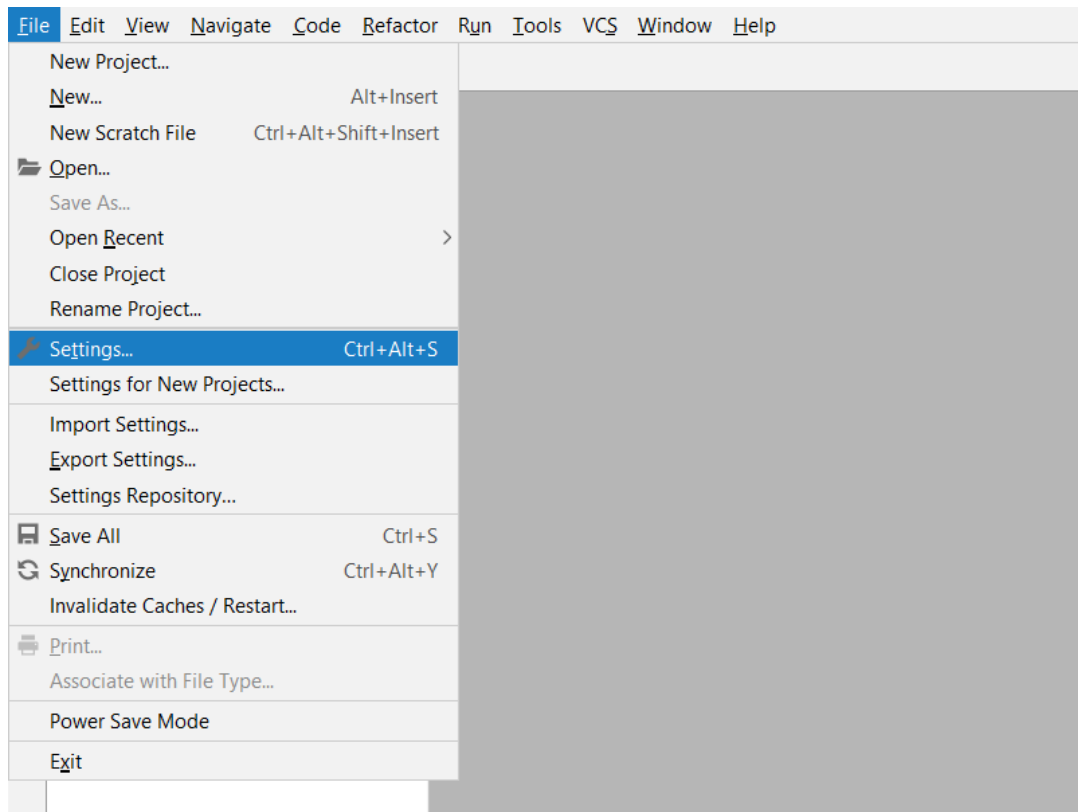
- OpenCV - názov balíčku: `opencv-python`
- Numpy - názov balíčku: `numpy`
- PIL - názov balíčku: `Pillow`
- Sklearn - názov balíčku: `scikit-learn`
- Scipy - názov balíčku: `scipy`

Na inštaláciu knižníc potrebných pre neurónovú sieť sme najprv nainštalovali Anacondu, ktorá je Open Sourceovou distribúciou pre jazyky Python a R pre vedecké výpočty.

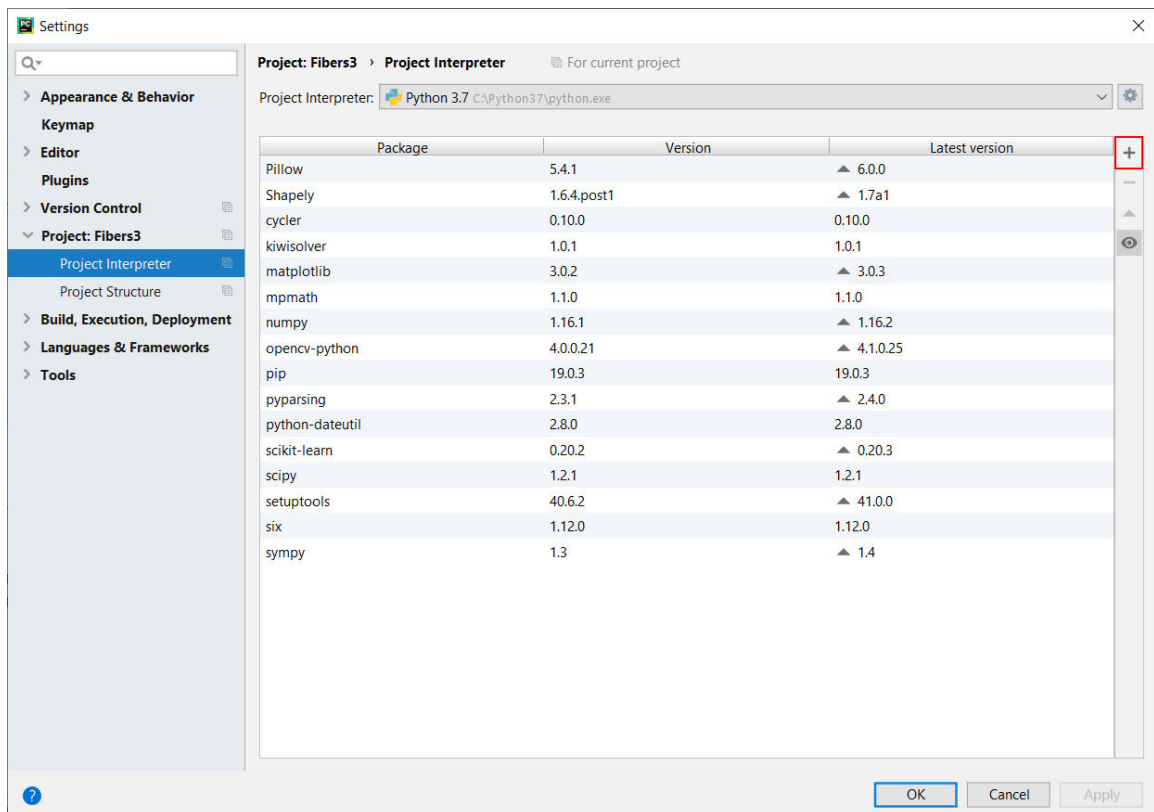
Anaconda - na inštaláciu sme stiahli inštaláčny súbor z oficiálnej stránky [anaconda.com](https://anaconda.com)

A pomocou príkazového riadku Anacondy (Anaconda Prompt) (Obrázok 46) sme nainštalovali knižnice:

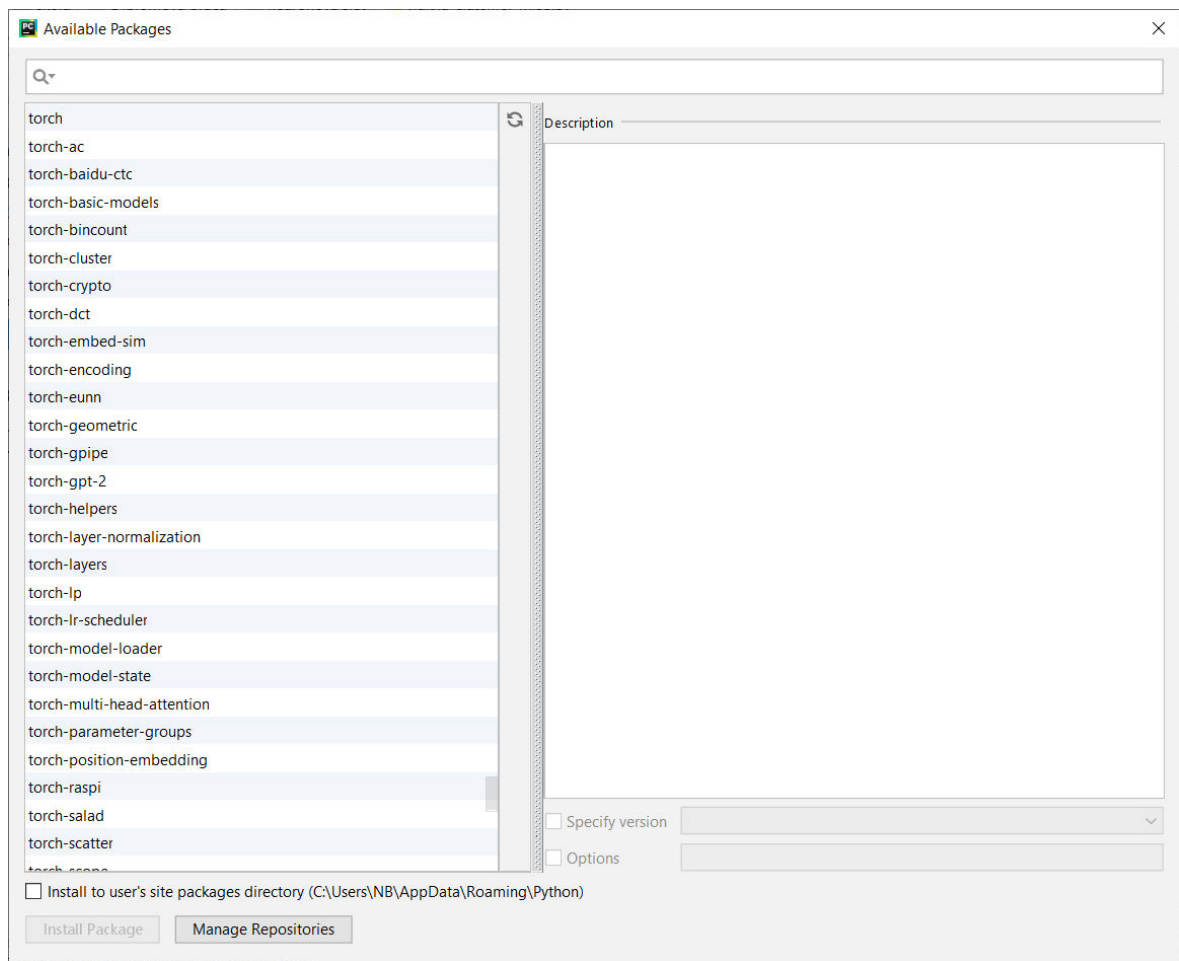
- Torch - spustením príkazu: `conda install pytorch -c pytorch`
- Torchvision – spustením príkazu: `pip install torchvision`



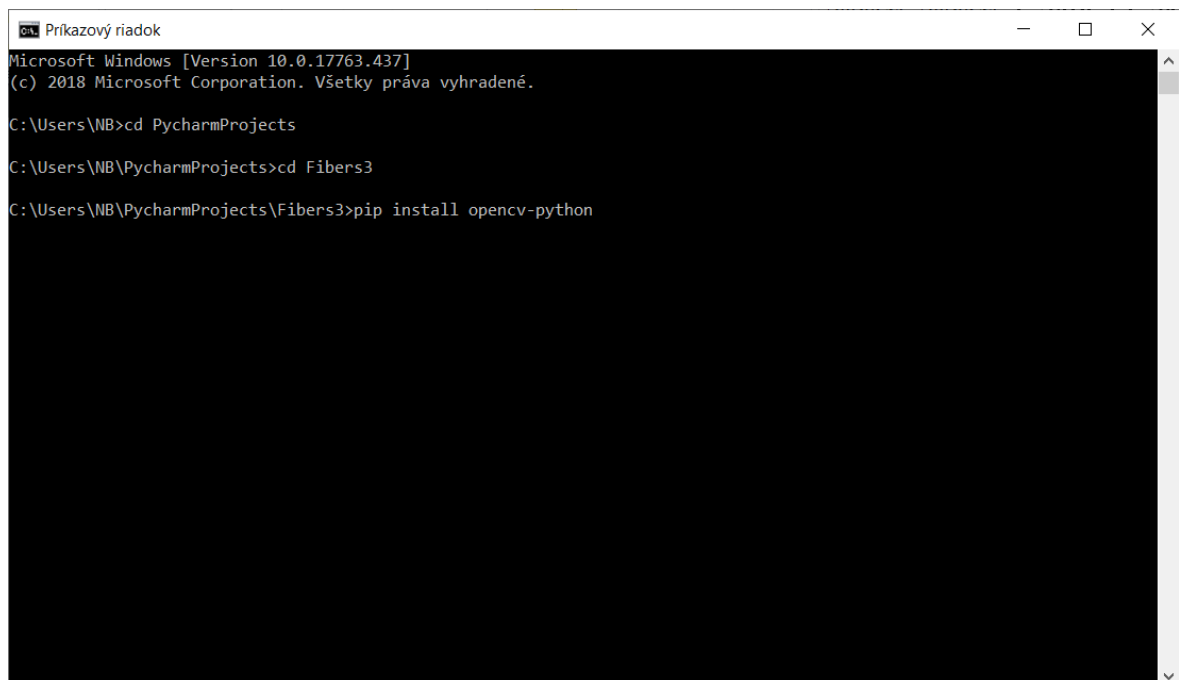
Obrázok 42 - PyCharm - Settings



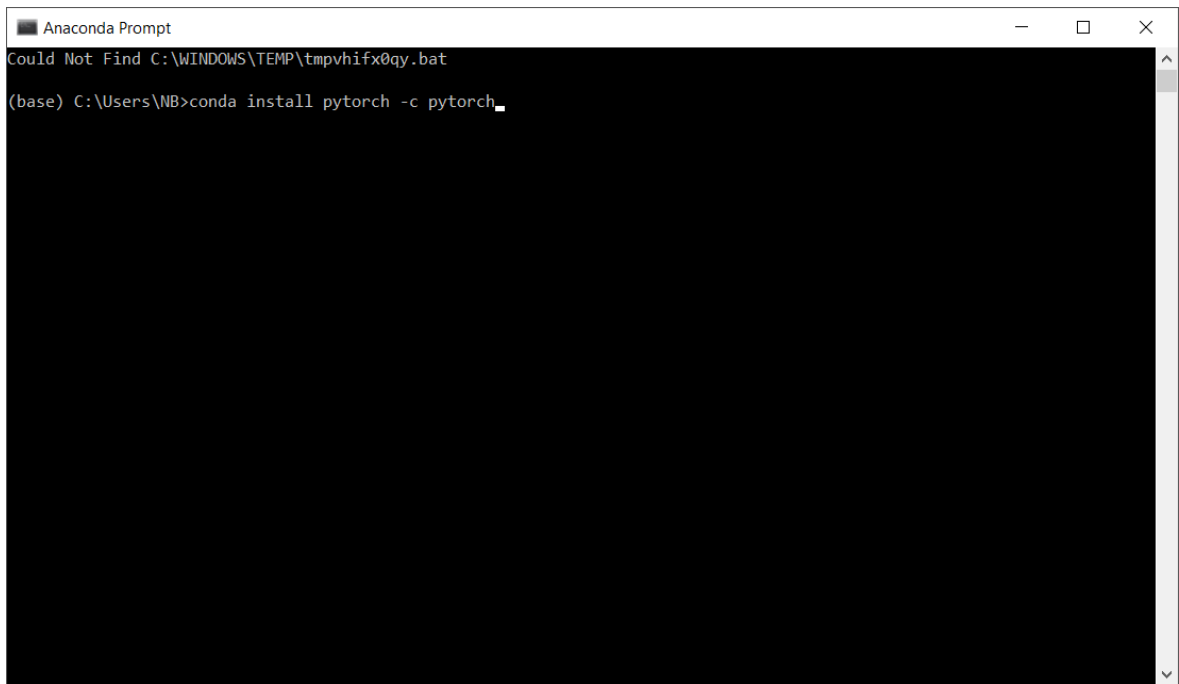
Obrázok 43 - PyCharm - Settings - Project Interpreter



Obrázok 44 - PyCharm - Settings - Project Interpreter - Available Packages



Obrázok 45 - Príkazový riadok



```
Anaconda Prompt
Could Not Find C:\WINDOWS\TEMP\tmpvhifx0qy.bat
(base) C:\Users\MB>conda install pytorch -c pytorch
```

Obrázok 46- Anaconda Prompt