

# Rozpoznávání objektů v textu

Cílem semestrální práce v předmětu Základy umělé inteligence je praktická implementace a použití algoritmů umělé inteligence. Doporučovány jsou zejména evoluční algoritmy. Jako téma semestrální práce jsem proto zvolil rozpoznávání objektů a textových útvarů v obrázcích. Hlavní motivací k tématu je možná využitelnost řešení v optických metodách rozpoznávání textů (OCR). Text v podobě obrázku je často žádoucí převést do upravitelné podoby, převod je ovšem výpočetně náročný a je výhodné použít paralelních algoritmů. Pokud jsme schopni převádět obrazový soubor paralelně, například po odstavcích, zvýší se celková efektivita převodu.

Principem řešení je implementace evolučních algoritmů přizpůsobených zvolenému tématu. Tyto algoritmy slouží k hledání globálního optima netriviálních úloh. Díky volbě tohoto přístupu stačí zvolit požadovanou kvalitu řešení, přičemž nás nemusí zajímat způsob, kterým výsledek získáme. Genetické algoritmy se vyznačují iteračním přístupem k řešení úlohy. Na začátku je zadána náhodná, případně heuristicky určená množina jedinců, na kterou se aplikují principy převzaté z evoluční biologie. Jedinec odpovídá možnému řešení úlohy, ve své datové části má zakódováno řešení problému a vyznačuje. Ke správné funkčnosti musí mít algoritmus hodnotící funkci, která určuje kvalitu jedince. Na množinu jedinců, se postupně aplikují metody křížení, mutace, a reprodukce. Během nich se na základě hodnoty fitness určí výhodnější řešení, která jsou v algoritmu zvýhodněna.

Můj přístup využívá geometrickou pravidelnost textu, jedinec je proto reprezentován množinou obdélníků, které odpovídají objektům ve vstupním souboru. Na počátku je náhodně inicializován, po proběhnutí základní heuristiky jsou obdélníky rozmístěny tak, aby se nepřekrývaly a nepřecházely mimo zadání, mají také omezenou velikost a nemohou rotovat. Tyto stavy by se daly případně penalizovat během výpočtu fitness, což ale vede ke zpomalení postupu hledání optima. Základní vlastností mého individuua je schopnost mutace, během které se může měnit i samotný počet genů. Genová mutace odpovídá translaci, škálování, spojování, dělení, odstranění a jejich kombinacím.

Hodnota fitness je počítána pro každý gen (obdélník) zvlášť. Velikost fitness se určí jako suma přes číselné hodnoty všech texelů. Celková kvalita řešení se určí jako vážený součet fitness genů podle jejich velikosti. Snažíme se vlastně vybrat, co množinu tmavých texelů v obrázku. Bylo by

ovšem nevhodné, kdyby v jinak dobrém řešení chyběl kus textu, přidal jsem proto threshold funkci, která silně penalizuje nezahrnutí texelů. Využití pouze této hodnoty by ale vedlo k nesmyslnému maximálnímu řešení, proto jsem zavedl druhou hodnotu, která jde proti fitness. Jde se o celkovou plochu řešení, kterou je nutno minimalizovat. Celkové řešení tedy hledá ekvilibrium mezi fitness a celkovou plochou.

Během průchodu algoritmem se opakují následující operace. Nejprve je na základně ruletového výběru zvolena dvě rodičovská řešení, z nich je potom vytvořena dvojice potomků. V závislosti na parametrech může proběhnout křížení, během něhož si dvojice vymění náhodné geny v jednobodovém křížení. Následně může proběhnou mutace, která náhodně upraví určitý počet genů pomocí již zmíněných metod. Po každém průběhu cyklem je vybráno řešení a největší fitness a je automaticky zahrnuto do dalšího řešení. Z důvodu složitosti křížení, je v průběhu důležitější samotná mutace.

Vylepšením naivního přístupu by mohla být forma učení na základně již vyřešených vstupních dat. V tomto módu by se vnitřní hodnoty automaticky nastavovaly a program by se mohl přizpůsobovat požadovanému výstupu. Nyní je nutné tyto parametry ručně určovat ještě před kompilací.

Mimo hlavní třídy algoritmu jsem vytvořil aplikaci ve frameworku Swing, která poskytuje přístup k základním parametrům a dovoluje vybírat zdrojový soubor. Zároveň aplikace zobrazuje postup řešení. Tím je zajištěno přibližné vizuální srovnání s předpokládaným optimálním řešením.