

Adam BLOKUS, Henryk KRAWCZYK
Politechnika Gdańska

POPRAWA JAKOŚCI BINARNYCH KLASYFIKACJI OBRAZÓW W NAGRA- NIACH WIDEO

Streszczenie. Rozpatrzono problem klasyfikacji obrazów występujących w sekwencji wideo. W tym celu wykorzystano dostępne algorytmy operujące na pojedynczych klatkach i zaproponowano oryginalną metodę ich udoskonalenia. Polega ona na analizie kolejnych fragmentów (okien) w rozpatrywanej sekwencji i polepszeniu wyników klasyfikatora pojedynczych obrazów na podstawie obserwowanych własności takich fragmentów. Przeprowadzone eksperymenty dla proponowanego podejścia potwierdzają poprawę jakości binarnych klasyfikacji oraz umożliwiają wskazanie wartości parametrów, dla których osiąga ono najlepsze rezultaty.

IMPROVING THE QUALITY OF BINARY CLASSIFICATIONS OF IMAGES IN VIDEO SEQUENCES

Summary. The work discusses the problem of classifying images in video sequences. Available algorithms operating on single frames are used and an original method of improving them has been proposed. It is based on analysing overlapping fragments (windows) of the video sequence and improving the quality of the original classifier by utilizing the properties of those fragments. The experiments which have been performed confirm an increase of the classification quality with the use of the new method and allow to point out values of parameters for which the best results are being acquired.

1. Wstęp

Wiele z popularnie stosowanych algorytmów do rozpoznawania cech i obiektów na pojedynczych obrazach jest wykorzystywanych do nagrań wideo (np. [8, 5, 21, 17, 7]). Algorytmy klasyfikacji obrazów używane w takich przypadkach stosowane są niezależnie do każdej z klatek, tak jak do pojedynczego obrazu. Metodę, w której stosuje się wybrany algorytm do kolejnych klatek w sekwencji wideo, oznaczamy dalej będziemy jako OFA (z ang. *One Frame Analyzed*). Jej główną zaletą jest szeroka dostępność różnych algorytmów klasyfikujących pojedyncze obrazy oraz odpowiednich zbiorów danych, na których tego rodzaju algorytmy można uczyć. Ponieważ każda z klatek rozpatrywana jest niezależnie od pozostałych, pomijana jest dodatkowa informacja wynikająca z sekwencji klatek, co jest główną wadą metod tego rodzaju. W niniejszej pracy zdefiniowana zostanie nowa metoda klasyfikacji wideo, usprawniająca tego typu tradycyjne podejście.

W przypadku wielu nagrań zaobserwować można oczywiste duże podobieństwo następujących po sobie klatek. Nagrania, które spełniają warunek zachodzenia małych zmian z klatki na klatkę, nazywać będziemy dalej nagraniami **ciągłymi**. Tym samym w każdym nagraniu istotne jest uwzględnienie w klasyfikacji nie tylko pojedynczej klatki, ale i jej najbliższego otoczenia. Zakres analizy klatek wyznacza tzw. okno czasowe, czyli symetryczne otoczenie danej klatki, zawierające tyle samo poprzedzających i następujących klatek.

W celu uniknięcia wysokich kosztów tworzenia wieloklatkowych zbiorów uczących, proponowane podejście oparte zostanie na istniejących klasyfikacjach OFA. Zostają one potraktowane jako sekwencja wartości binarnych, którą kolejny krok algorytmu przetworzy z uwzględnieniem właśnie zależności czasowych. Binarny charakter klasyfikacji oznacza, że rozpatrywane są algorytmy OFA takie, które przypisują wejściowe obrazy do jednej z dwóch kategorii. Na koniec niniejszej publikacji przedstawione zostaną propozycje rozszerzenia takiego podejścia na inne rodzaje problemów klasyfikacyjnych.

W pracy [5] wprowadzony został algorytm śledzenia twarzy, który wykrywa twarze jako zestaw statycznych cech, oddzielnie w każdej klatce wejściowego nagrania wideo. Następnie trajektorie twarzy zostają skorygowane pod kątem możliwych błędów klasyfikacji i wygładzone za pomocą filtru Kalmana. Samo wygładzanie jest metodą stosowaną w różnych zastosowaniach, w których otrzymana krzywa albo sekwencja musi być przekształcona w bardziej prawdopodobną formę. Przykładowo, w pracy [9] rozpoznana ścieżka ruchu dłoni zostaje wygładzona za pomocą algorytmu, który koryguje błędy we wstępnej sekwencji pozycji dłoni. Odstające pozycje w sekwencji zostają odfiltrowane i zastąpione bardziej prawdopodobnymi, wynikającymi z interpolacji krzywej z otaczających pozycji sekwencji.

Praca [1] wykorzystuje własność ciągłości obrazu wideo dla zbiorowego przydzielania wspólnej klasyfikacji ciągom podobnych kolejnych klatek. Metoda ta pozwala na przyspieszenie klasyfikacji całej sekwencji w przypadku kosztownego obliczeniowo algorytmu klasyfikującego pojedyncze klatki.

W pracy [10] przedstawiony został szeroki przegląd i analiza zastosowania metod wygładzających sekwencje za pomocą ruchomego okna, którego centralny element zastępowany jest odpowiednio ważoną sumą wszystkich elementów w tym oknie. Podobnego rodzaju pomysł zastosowany został w przetwarzaniu sekwencji klasyfikacji w nagraniu audio. W pracy [18] po algorytmie rozpoznającym występowanie wokalu na nagraniu zastosowane zostaje wygładzanie wstępnie otrzymanych rezultatów. Zastosowana metoda cechuje się bardziej całościowym podejściem, w którym sekwencja klasyfikacji traktowana jest jako ciąg sygnałów wyjściowych w ukrytym modelu Markowa. Otrzymana za pomocą algorytmu Viterbiego sekwencja stanów ukrytych cechuje się dużo mniejszą fragmentacją niż wstępne wyniki, jak i zwiększoną w porównaniu z nimi jakością klasyfikacji.

Praca [20] przedstawia przegląd algorytmów i analizę dziedziny przestrzenno-czasowego (ang. *spatio-temporal*) wykrywania obiektów i zdarzeń w nagraniach wideo. Przedstawione zostały liczne metody, w których klasyfikacja wideo uwzględnia nie tylko własności obiektów i zdarzeń na pojedynczych obrazach, ale również ich wzajemne położenia w czasie.

Jednym z przykładów intuicyjnego zastosowania metody wygładzania sekwencji wyjściowej jest praca [3]. Zastosowano w niej symetryczne okno z pięcioma klatka-

mi do korygowania wstępnej sekwencji klasyfikacji (wykrycia obecności przyrządów endoskopowych w obrazie z badania). Jedynie pobieżnie wspomniane zostały próby z innymi rozmiarami okna. Również autorzy [6] zastosowali w opisywanym algorytmie dodatkowy krok przetwarzania końcowego (ang. *post-processing*), w którym wykorzystując większościowe głosowanie w oknie o rozmiarze 5 klatek ustalona zostaje końcowa wartość środkowego elementu. Autorzy bez dodatkowej analizy sugerują, że taki krok poprawia skuteczność ich metody.

W pracach [12] oraz [13] zaproponowany został algorytm do wykrywania kolistych obszarów, zawierających właściwą treść w nagraniach z kamer endoskopii kapsułkowej. Ze względu na wykorzystanie specyficznych kamer, rejestrowany obraz z kamery zawiera się w okręgu i jest przedstawiony w poszczególnych klatkach na czarnym tle. Całość pokrywana jest w różny sposób napisami oraz symbolami kontrolnymi. Wyniki algorytmu zostają na koniec przetworzone tak, aby wyeliminować odstające klasyfikacje i zapewnić zgodną klasyfikację dla sekwencji co najmniej 100 klatek. Tym samym klasyfikacja każdej pojedynczej klatki zostaje uzależniona od wielu poprzedzających i następujących klatek.

Cox i Snell, w swojej książce o danych binarnych ([4]) przedstawiają problemy tzw. binarnych serii czasowych. W przypadku binarnych klasyfikacji klatek nagrania wideo (tzn. przy ich podziale na dwie klasy), rezultatem są właśnie binarne sekwencje czasowe, wobec których wykorzystać można metody analizy przedstawione w tej książce. Wśród nich najbardziej cenne dotyczą: zamodelowania ciągu klasyfikacji jako realizację procesu markowskiego dla oszacowania prawdopodobieństwa poszczególnych sekwencji oraz potraktowania binarnych wartości jako wartości funkcji gausowskiej zmiennej losowej.

Zaproponowana w następnych rozdziałach metoda bazuje na algorytmach OFA oraz wykorzystaniu efektywnej metody analizy okien czasowych w otrzymanym ciągu binarnym. Rozpatrzone zostaną oryginalne metody jego wygładzania w celu zwiększenia jakości rozpoznania całej sekwencji obrazów.

Przyjmuje się, że algorytm OFA na wejściu analizuje pojedyncze klatki i dla każdej z nich zwraca binarną wartość klasyfikowanej cechy. W dodatkowym kroku wykonywana jest korekta klasyfikacji na podstawie okna złożonego z kilku klatek, co prowadzi do nowych metod analizy FSA (z ang. *Frame Sequence Analysis* - analiza sekwencji klatek).

2. Metoda analizy OFA

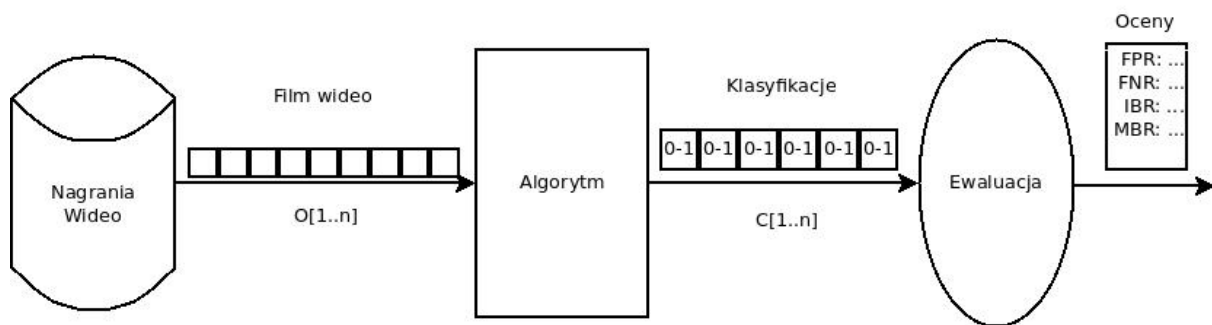
Rysunek 1 przedstawia uproszczony schemat algorytmu klasyfikującego obrazy w sekwencji wideo, którym jest algorytm typu OFA. Strumień klatek z danych wideo trafia na wejście algorytmu, który zwraca w odpowiedzi sekwencję klasyfikacji. Otrzymana sekwencja jest oceniana pod kątem czterech kryteriów określających ogólną poprawność klasyfikacji oraz poprawność podziału na sceny.

Przez OFA_x oznaczamy będziemy x -ty algorytm tego typu wykorzystany w analizie klatki. Algorytmy typu OFA wspomniane powyżej, wykorzystywane są w zastosowaniach takich jak np.: wykrywanie krwawień i zmian chorobowych w nagraniach endoskopii kapsułkowej, wykrywanie twarzy i sylwetek, śledzenie obiektów czy zliczanie pojazdów w ruchu ulicznym. Z definicji algorytmu OFA wynika, że może on powstać

na bazie dowolnego algorytmu klasyfikującego obrazy bądź wykrywającego cechy lub obiekty w obrazie. Do tego rodzaju metod należą np. algorytmy wykrywające obiekty na podstawie punktów charakterystycznych (np. [19]) bądź klasyfikujące obrazy na podstawie cech tekstury ([15]).

Wybrany przez nas do eksperymentu przykładem takiego algorytmu jest również algorytm wykrywania obiektów za pomocą kaskad opartych na prostych cechach (z ang. *Cascades of simple features* [21, 17]). W tym celu wykorzystuje się wyuczony klasyfikator detekcji sylwetek, dystrybuowany wraz z implementacją algorytmu w bibliotece OpenCV ([2]). Wyszukiwanie za pomocą kaskad szczególnie stosowane jest w przypadku poszukiwania obiektów o schematycznych charakterystykach (np. twarzy, sylwetki bądź znaku). Za pomocą łańcucha prostych klasyfikatorów o przeciętnych parametrach uzyskuje się zwiększenie jakości wykrycia, które w przypadku eksperymentów rozpatrywanych w literaturze osiąga dokładność rzędu 80-100%.

Algorytm wykrywania za pomocą kaskad na wejściu przyjmuje tylko jeden obraz i po jego analizie stwierdza obecność bądź brak ludzkiej sylwetki w kadrze. Jest więc algorytmem OFA dokonującym binarnej klasyfikacji (0 - brak sylwetki, 1 - istnienie sylwetki).



Rys. 1. Schemat działania algorytmu klasyfikacji wideo.

W przypadku algorytmu OFA, złożoność obliczeniowa określona jest jako iloczyn liczby klatek w klasyfikowanym wideo ze złożonością klasyfikacji pojedynczej klatki, która jest funkcją zależną m.in. od parametrów obrazu. Złożoność obliczeniową określonego algorytmu OFA_x możemy więc określić jako

$$OFA_x \sim O(n \cdot f_{OFA_x}(D)) \quad (1)$$

gdzie D - parametry obrazu, $f_{OFA_x}(D)$ - funkcja wyrażająca złożoność klasyfikacji pojedynczego obrazu.

W przypadku ustalonych lub ograniczonych parametrów obrazu mamy $D \sim O(1)$, co prowadzi do ostatecznej praktycznej złożoności

$$OFA_x \sim O(n) \quad (2)$$

Jakość algorytmu określona jest czterema wartościami, określającymi różne rodzaje błędów zaistniałych w wynikach klasyfikacji. Błędy te wynikać mogą z niespełnienia dwóch oczekiwań wobec algorytmów klasyfikujących nagrania wideo. Po pierwsze, oczekiwane jest osiągnięcie wysokiego współczynnika poprawności klasyfikacji pojedynczych klatek. Do pomiaru liczby błędów w tej kategorii wykorzystane zostaną dwa

współczynniki - tzw. błędy pierwszego i drugiego rodzaju (ang. *false positive ratio* - FPR i *false negative ratio* - FNR). FPR określa się jako stosunek liczby klatek negatywnych (nie posiadających poszukiwanej cechy) niewłaściwie sklasyfikowanych jako pozytywne (posiadające pewną cechę) do liczby wszystkich klatek negatywnych, zaś FNR jako stosunek liczby klatek pozytywnych niewłaściwie sklasyfikowanych jako negatywne do liczby wszystkich klatek pozytywnych. Parametry FPR i FNR określone są więc następującymi wzorami:

$$FPR = \frac{\#(\text{Klatki negatywne sklasyfikowane jako pozytywne})}{\#(\text{Klatki negatywne})} \quad (3)$$

$$FNR = \frac{\#(\text{Klatki pozytywne sklasyfikowane jako negatywne})}{\#(\text{Klatki pozytywne})} \quad (4)$$

(# oznacza moc danego zbioru).

W przypadku klasyfikacji nagrań wideo i późniejszej prezentacji wyników pojawia się potrzeba podzielenia wejściowego nagrania na tzw. sceny, czyli dłuższe sekwencje klatek o jednakowym wyniku klasyfikacji. W przypadku rzeczywistych nagrań tego rodzaju oczekiwanie jest zasadne, gdyż pojawienie się każdego obiektu lub cechy w kadrze wiąże się ze stopniowym wejściem (np. wynurzenie z oddali, wejście spoza kadru, wyjście zza zasłaniającego obiektu, zmiana sytuacji w kadrze), które wiąże się z wydłużoną widocznością poszukiwanej cechy. Ciągi klasyfikacji, których własności pokrywają się z tego rodzaju własnościami rzeczywistych zjawisk, lepiej oddają charakter rozpoznawanych zjawisk. Dodatkowo, w niektórych zastosowaniach prawidłowy podział na sceny decyduje o użyteczności otrzymanych wyników (np. w endoskopii kapsułkowej [8] mocna fragmentacja klasyfikacji sprawi, że otrzymane wyniki nie przyśpieszą pracy lekarza, który de facto wciąż będzie musiał drobiazgowo analizować całość filmu klatka po klatce). Dla tego rodzaju kryterium istotne są więc granice między scenami - czyli miejsca, w których zmienia się wynik klasyfikacji w sekwencji (z 0 na 1 bądź z 1 na 0).

Aby zrealizować drugi wymóg wobec jakości otrzymywanej klasyfikacji, zdefiniowane zostają miary podobne do FPR i FNR, oparte na poprawności umieszczenia granic między scenami. Pierwsza z nich, IBR (z ang. *Invalid Boundaries Ratio* - współczynnik niewłaściwych granic), określa stosunek niewłaściwych przejść między scenami do wszystkich przejść między scenami. Określony został w następujący sposób:

$$IBR = \frac{\#(\text{Niewłaściwe granice między scenami})}{\#(\text{Wykryte granice między scenami})} \quad (5)$$

Drugi ze współczynników to MBR (z ang. *Missed Boundaries Ratio* - współczynnik pominiętych granic). Określa on stosunek liczby niedopasowanych granic między scenami do faktycznej liczby wszystkich granic między scenami w sposób następujący:

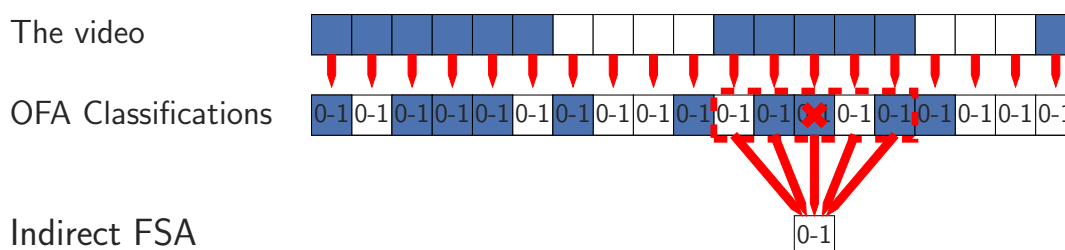
$$MBR = \frac{\#(\text{Pominięte granice między scenami})}{\#(\text{Faktyczne granice między scenami obrazu wejściowego})} \quad (6)$$

W ciągłych nagraniach występuje zjawisko częściowej widoczności wykrywanych obiektów lub cech (w trakcie np. wyłaniania się z oddali bądź wchodzenia w kadr), w trakcie której nie jest możliwe jednoznaczne ustalenie wartości klasyfikacji (np. stwierdzenie w jakim stopniu widoczna powinna być sylwetka, aby stwierdzić, że w kadrze znajduje się człowiek). W związku z tym dopuszczalna jest pewna tolerancja

w przypadku ustalania poprawności umieszczenia granicy, która określona będzie przez parametr L - liczbę pozycji, o jaką mogą się różnić granice w ocenianej klasyfikacji, aby wciąż jeszcze zostały uznane za zgodne z wzorcem¹. Miary będziemy oznaczać odpowiednio jako IBR_L , MBR_L , jeżeli z kontekstu nie będzie jasno wynikała obrana wartość L .

3. Metoda analizy FSA

Poniżej zaproponowane zostaje nowe podejście FSA. W przeciwieństwie do klasycznych algorytmów OFA, gdzie analizowana jest jedna klatka, ostateczna klasyfikacja każdej klatki oparta jest nie tylko na wyniku jej indywidualnej klasyfikacji, ale również na klasyfikacjach klatek ją otaczających. W podejściu FSA klasyfikator OFA stosowany jest jedynie w pierwszym etapie. Drugim etapem jest końcowe przetworzenie otrzymanych wstępnych wyników, z uwzględnieniem oczekiwanych charakterystyk zmienności rozpoznawanej cechy. Ilustracja tego podejścia przedstawiona została na Rysunku 2. Każda z klatek obrazu wideo ma pewną ukrytą, nieznaną algorytmowi binarną wartość rozpoznawanej cechy. Załóżmy, że po wstępnej klasyfikacji na zaznaczonej pozycji udzielona została niepoprawna odpowiedź. Analiza całego okna (oznaczonego przerywaną linią) pozwala na zmianę tej wartości (np. głosowanie większościowe) poprzez uwzględnienie sąsiednich wartości w czasie kolejnej klasyfikacji.



Rys. 2. Ilustracja działania metody FSA.

Postępowanie w drugim etapie algorytmu FSA może być zdefiniowane za pomocą metod statystycznych, interpolacji, jako proces losowy czy też jako korekcja sygnału o określonych własnościach i szumie. Na Rysunku 3 zaprezentowany został pseudokod proponowanego algorytmu FSA. W przeciwieństwie do przykładów stosowania podobnego podejścia w literaturze, rozmiar okna i próg akceptacji głosowania nie są ustalone i występują jako parametry. Dodatkowo, w postaci malejących geometrycznie wag, uwzględniona jest odległość klatek w oknie od klatki środkowej i związane z nią stopnie coraz mniejsze oczekiwane powiązanie wyników klasyfikacji.

Rozpatrywany algorytm $FSA_{w,\lambda,A}$ operuje na sekwencji binarnej o długości n , przesuwając okno o długości w . W postaci zaproponowanej w pseudokodzie złożoność obliczeniowa $FSA_{w,\lambda,A}$ wynosi $\sim O(nw)$.

¹Warto zwrócić uwagę, że niezależnie od wartości parametru, obie miary wciąż określają jakość klasyfikacji, jednak zmienia się jej rozumienie. W przypadku $L = 0$ określana jest idealna zgodność z wzorcem, przy niewielkich wartościach L nieznaczne przesunięcia względem niego, natomiast przy $L \rightarrow \infty$ miary określają zgodność liczby scen.

Algorytm FSA_{w,λ,A}:

Wejście: Sekwencja klasyfikacji OFA $O[1 \dots n]$ o długości n otrzymana z zastosowania algorytmu OFA_x ($O \in \{0, 1\}^n$)

Wyjście: Sekwencja klasyfikacji końcowych FSA $C[1 \dots n]$ o długości n ($C \in \{0, 1\}^n$)

Parametry:

- w - szerokość okna, $5 \leq w \leq w_{\max}$
- $0 < \lambda \leq 1$ - współczynnik rozkładu istotności klatek w oknie czasowym
- próg akceptacji głosowania $0,5 < A \leq 1$

Algorytm:

1. $v = \lfloor \frac{w}{2} \rfloor$ // liczba klatek po jednej stronie okna ($w = 2v + 1$)
2. $W = [\lambda^v \dots, \lambda, 1, \lambda, \dots, \lambda^v]$ // rozkład istotności klatek
3. $C = O$ //inicjalizacja obliczeń
4. for i in $1 + v \dots n - v$
5. $O_{i,w} = O[i - v \dots i + v]$ // wybór okna
6. $c = \frac{\sum_{j=1}^w W[j] O_{i,w}[j]}{\sum_{j=1}^w W[j]}$ // głosowanie
7. if $c \geq A$: $C[i] = 1$ //decyzja o zmianie klasyfikacji
8. elif $c \leq 1 - A$: $C[i] = 0$
9. return C

Rys. 3. Pseudokod algorytmu FSA

Całkowita klasyfikacja obrazu (złożenie OFA_x i FSA_{w,λ,A}) ma więc złożoność czasową $\sim O(n \cdot f_{\text{OFA}_x}(D) + n \cdot w) = O(n(f_{\text{OFA}_x}(D) + w))$

Wartość w ograniczona jest poprzez warunek zachowania podobieństwa przez klatki znajdujące się w obrębie jednego okna i dla danej kategorii wideo ograniczona jest przez określoną stałą wartość. W sensie asymptotycznym zastosowanie schematu FSA nie zwiększa więc złożoności obliczeniowej tego algorytmu. Podobnie, dla danego zastosowania strumień wejściowego wideo ma określone, ustalone parametry obrazu (rozdzielczość i głębia kolorów). Możemy więc ustalić zarówno $w \sim O(1)$, jak i $D \sim O(1)$, uzyskując:

$$\text{OFA}_x + \text{FSA}_{w,\lambda,A} \sim O(n) \quad (7)$$

Głównym celem zastosowania schematu FSA jest zwiększenie jakości dokonywanej klasyfikacji, na którą wpływ ma ustalenie odpowiednich wartości parametrów sterujących. Przedmiotem zainteresowania staje się dobór:

- w - szerokości okna,
- λ - współczynnika z jakim maleje waga głosów wraz z odległością od centrum okna,
- A - progu, który musi być przekroczony, aby głosowanie przeważało na stronę danej wartości

taki, aby zminimalizować oceny FPR, FNR, IBR i MBR.

Aby uniezależnić prowadzone poszukiwania od jakości poprawianej funkcji OFA, wybierzemy następujące podejście do problemu optymalizacji : Poszukujemy takiego schematu FSA, który jest najlepszy pod względem liczby błędów algorytmu OFA, które niweluje. Dla każdego kryterium c ze zbioru $C = \{FPR, FNR, IBR, MBR\}$ i danych $FSA_{w,\lambda,A}$ i OFA_x określamy tę proporcję współczynnikiem:

$$r_c(w, \lambda, A, x) = \frac{c(FSA_{w,\lambda,A})}{c(OFA_x)} \quad (8)$$

Przy powyższej definicji np. $r_{FNR}(w, \lambda, A, x) = 0,7$ oznaczałoby, że zastosowanie algorytmu $FSA(w, \lambda, A)$ usunęło 30% początkowej liczby błędów typu false negative (otrzymanych w wyniku zastosowania jedynie algorytmu OFA_x).

Dla zbiorczej oceny całkowitej jakości algorytmu przyjmujemy funkcję Q_{FSA} jakości algorytmu FSA:

$$Q_{FSA}(w, \lambda, A) = \sqrt{\sum_{c \in C} r_c^2(w, \lambda, A, x)} \quad (9)$$

Osiąga ona wartość 0 (najlepszą) tylko wtedy, gdy wszystkie z czterech miar jakości równe są zeru (czyli nie ma żadnych błędów w klasyfikacji). Pod uwagę brana jest względna ilość zachowanych błędów OFA_x . Ponieważ zastosowana jest norma l_2 wektora współczynników poszczególnych kryteriów, wartość funkcji jakości zdominowana jest przez największą względną wartość, co wymusza harmonijne optymalizowanie wszystkich czterech kryteriów. (W przypadkach, w których w trakcie eksperymentu okazało się, że któreś z wartości nie da się poprawić, w dalszej części dopuszczone zostało pominięcie jednego z kryteriów przy liczeniu normy).

Podstawową sprawą jest odpowiednia parametryzacja algorytmu FSA dla danego OFA_x , aby otrzymać optymalny $FSA_{w,\lambda,A}$. Stąd istotne staje się poszukiwanie najlepszego zestawu parametrów dla zaproponowanej metody FSA, które sprowadza się do rozwiązania następującego problemu:

Problem 1.

Dla jakich wartości parametrów w, λ, A algorytmu $FSA_{w,\lambda,A}$ opartego na danym OFA_x osiąga swoje minimum funkcja:

$$Q_{FSA}(w, \lambda, A)$$

gdzie zachodzą następujące ograniczenia:

- $5 \leq w \leq w_{max}$
- $0 < \lambda \leq 1$
- $0,5 < A \leq 1$

Do rozwiązania przedstawionego problemu zostanie zastosowana przybliżona metoda dyskretnej optymalizacji parametrów λ i A połączona z całościowym przeszukiwaniem przestrzeni dopuszczalnych wartości parametru w . Schemat działania metody optymalizacji przedstawiony został na rysunku 4. Wykorzystana metoda optymalizacji definiuje zachowanie algorytmu w wierszach 3 do 13. Określający ją wyznacznik OPTIMIZER wyznacza trzy metody:

- `initOPTIMIZER` - określającą punkt startowy poszukiwań
- `criteriumOPTIMIZER` - sprawdzającą kryterium zbieżności
- `nextOPTIMIZER` - wyznaczającą kolejny testowany punkt na podstawie dotychczasowych wyników.

Schemat algorytmu optymalizacji parametrów w, λ, A : Wejście:

- `limits` - zestaw ograniczeń na wyznaczane parametry
 - w_{min}, w_{max} - minimalna i maksymalna wartość w (nieparzyste, całkowite)
 - ograniczenia na wartość parametrów λ, A wynikają z ich znaczenia i określone są jako $0 < \lambda \leq 1$ oraz $0,5 \leq A \leq 1$
- `max_iter` - ograniczenie liczby iteracji
- `threshold` - kryterium zakończenia optymalizacji

Wyjście: $w_{OPT}, \lambda_{OPT}, A_{OPT}$ - najlepsze znalezione wartości

Parametry: `OPTIMIZER` - metoda optymalizacji określająca funkcje: `initOPTIMIZER`, `criteriumOPTIMIZER`, `nextOPTIMIZER`

Algorytm:

```

1.  $q_{min} = \infty$  # inicjalizacja minimum
2. for  $w$  in  $w_{min}..w_{max}$  step 2:
3.    $results_w = []$  # lista wyników
4.    $\lambda_0, A_0 \leftarrow initOPTIMIZER(limits)$  # inicjalizacja wyszukiwania
5.   for  $i$  in  $0..max\_iter$ :
6.      $q_{current} = Q(w, \lambda_i, A_i)$ 
7.      $results_w += (q_{current}, w, \lambda_i, A_i)$  # zapamiętanie wyniku
8.     if  $q_{current} < q_{min}$ : # zapamiętanie minimum
9.        $q_{min} = q_{current}$  # zapamiętanie minimum
10.     $w_{OPT}, \lambda_{OPT}, A_{OPT} = w, \lambda_i, A_i$ : # zapamiętanie minimum
11.    if criteriumOPTIMIZER( $results_w, threshold$ ):
12.      break # koniec iteracji, jeżeli metoda osiągnęła minimum
13.     $\lambda_{i+1}, A_{i+1} \leftarrow nextOPTIMIZER(limits, results_w)$  # wybór kolejnego punktu
14. return  $w_{OPT}, \lambda_{OPT}, A_{OPT}$ 

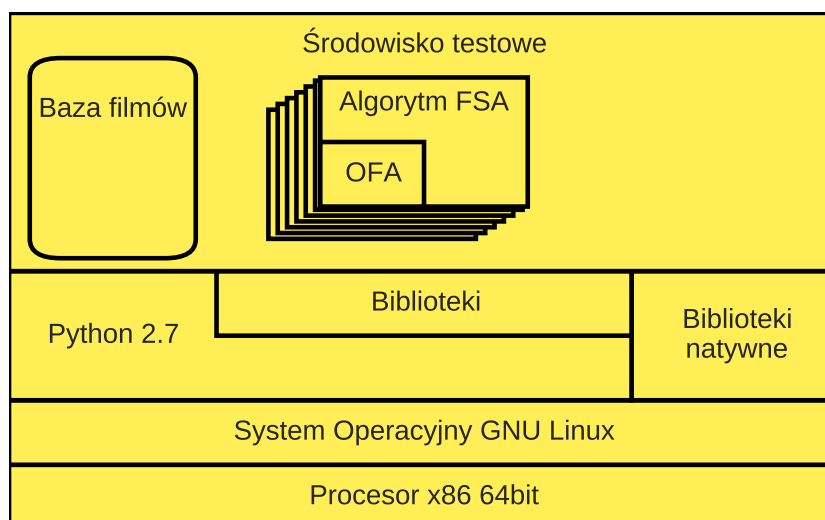
```

Rys. 4. Pseudokod algorytmu optymalizującego wartości parametrów w, λ, A .

Rozważana metoda ma trzy parametry zawierające się w ograniczonych przedziałach, z czego jeden (rozmiar okna w) jest dyskretny i ma względnie niewielki zbiór dopuszczalnych wartości. Dla każdej z nich prostokąt wyznaczony przez dopuszczalne zakresy λ i A można pokryć gęsto losowymi punktami w tzw. podejściu Monte Carlo - czyli losowym przeszukaniu dostępnych wartości parametrów. Dzięki temu nie jest konieczne stosowanie którejś z bardziej złożonych metod optymalizacji dostępnej w bibliotece do obliczeń naukowych SciPy[16] języka Python (np. L-BFGS-B [11] bądź truncated Newton algorithm [14]).

4. Procedura testowania

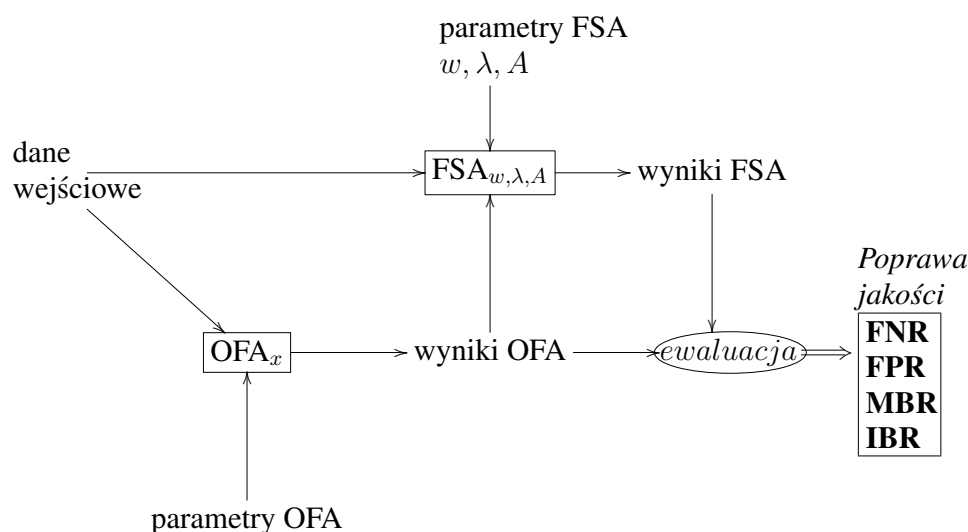
Celem wprowadzonej metody FSA jest poprawienie jakości algorytmów OFA. Zatem po wyznaczeniu wszystkich czterech współczynników błędów (wzory 3-6) dla algorytmu OFA i opartego na nim algorytmu FSA, ocenić można w jakim stopniu takie podejście faktycznie poprawia jakość otrzymywanej klasyfikacji.



Rys. 5. Schemat środowiska testowego.

Do celu testów wybrane algorytmy zostały zaimplementowane w przygotowanym środowisku testowym opartym na środowisku języka Python 2.7 oraz narzędzi i bibliotek do odczytu obrazu, operacji obliczeniowych oraz uczenia maszynowego. Na rysunku 5 przedstawiony został schemat systemu testowego, w którym uruchomione zostało środowisko testowe. Ze względu na zastosowanie środowiska Python, w bardzo dużej mierze niezależnego od środowiska sprzętowego, faktyczny wpływ na wyniki eksperymentu mają jedynie algorytmy zaimplementowane w ramach niniejszej pracy oraz te wykorzystane z bibliotek.

Środowisko uruchomieniowe Python wprowadza dodatkową warstwę pośrednią między aplikacjami użytkownika a znajdującym się pod nią systemem operacyjnym. Ustandaryzowany kod bajtowy (ang. *bytecode*) oraz reguły jego wykonania niezależnie od systemu i sprzętu, zapewniają przenośność osiągniętych wyników jakościowych. Od konfiguracji sprzętowej w związku z tym zależna pozostaje jedynie wydajność testowanych algorytmów, która nie jest przedmiotem niniejszej publikacji.

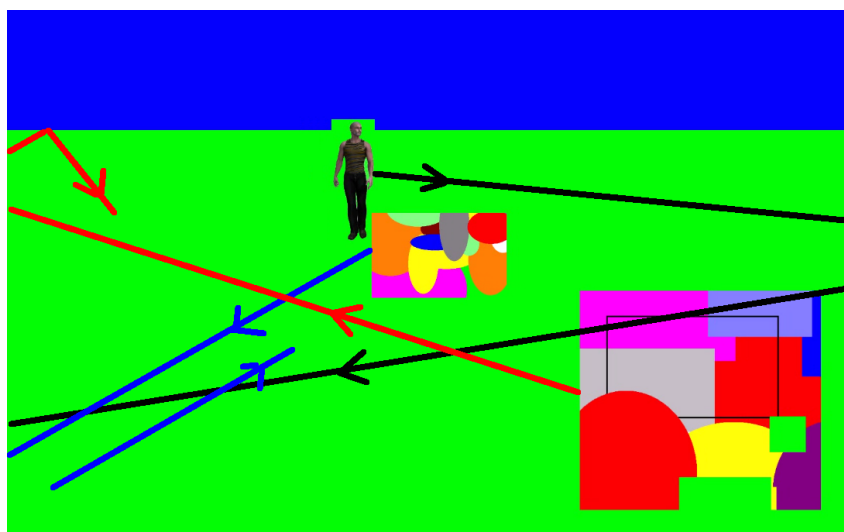


Rys. 6. Procedura testowa oceny jakości klasyfikacji OFA oraz FSA

Ogólny schemat samej procedury testowej zaprezentowany został na Rysunku 6. Wejście każdego testu stanowią trzy elementy:

1. dane wideo (ze znaną z góry poprawną klasyfikacją) w postaci nagrań z sekwencjami poprawnych klasyfikacji,
 2. parametry bazowego algorytmu OFA,
 3. parametry zastosowanego algorytmu FSA wraz z doбором funkcji pomocniczych,
- a także:
4. testowany algorytm FSA oraz bazowy algorytm OFA.

Dane wejściowe (nagranie wideo reprezentujące ruchome obiekty) są wpiery używane do uzyskania wstępnych klasyfikacji za pomocą algorytmu OFA ("wyniki OFA" na rys. 6). Następnie, wraz ze wstępnymi klasyfikacjami trafiają do algorytmu FSA, który zwraca klasyfikacje końcowe ("wyniki FSA"). Jakość algorytmu FSA jest oceniana poprzez porównanie klasyfikacji końcowych z wstępnymi pod kątem czterech wspomnianych wcześniej kryteriów (FPR, FNR, IBR, MBR). W przypadku miar IBR oraz MBR wykorzystany został parametr L o wartości 5, 10, 20 i 50, co oznacza tolerancję na zmianę klasyfikacji odległą o maksymalnie L pozycji od wartości wzorcowej. W testowanym nagraniu (25fps) odpowiada to 0,2-2s, co jest wartością znacznie mniejszą lub zbliżoną do czasu wejścia w kadr przy prędkości, z jaką przemieszczają się obiekty na obrazie.



Rys. 7. Trajektorie ruchu obiektów na klatce testowego strumienia wideo.

W niniejszej pracy ocena jakości algorytmów FSA dokonana została na dwóch rodzajach danych. Pierwszym z nich jest modelowy strumień, generowany w sztuczny sposób. Algorytm wykrywania sylwetek za pomocą kaskad zastosowany został do wykrywania postaci widocznej w kadrze wśród innych ruchomych obiektów. W zależności od położenia, zgodnie z zasadami perspektywy, zmienia się postrzegany rozmiar obiektów, w tym poszukiwanej sylwetki. Ruchome obiekty przemieszczają się po polu widzenia oraz w obszarach za jego granicami, "odbijając" się od granic częściowo widocznego pola. Sylwetka wraz z zasłaniającymi obiektami i szkice ich trajektorii przedstawione

zostały na rysunku 7. W przypadku zbliżonego położenia jednego z obiektów zasłaniających do poszukiwanej sylwetki, sylwetka zostaje częściowo lub w całości zakryta, co zakłóca wyniki klasyfikacji. Modelowy strumień występuje w dwóch wariantach - oryginalnym oraz z zakłóceniami. Wśród stosowanych zakłóceń są zarówno takie, które na dłuższy czas zniekształcają obraz (np. rozmycie) oraz takie, które są jedynie skokowo pojawiającymi się artefaktami.

Drugim rodzajem danych jest zbiór Chokepoint [22]. Są to opisane nagrania osób chodzących po korytarzach, przygotowane dla zastosowania w eksperymentach dotyczących rozpoznawania i identyfikacji ludzi. Organizacja NICTA², która je opracowała, udostępnia je bezpłatnie do celów niekomercyjnych badań naukowych³. Podobnie jak w poprzednim przypadku, do klasyfikacji wykorzystany został dostępny w bibliotece OpenCV algorytm odnajdujący twarze na obrazach. Wynikiem opartego na nim algorytmu OFA jest obecność jakichkolwiek twarzy w obrazie, bądź ich brak.

W każdym przypadku testowym zarówno do optymalizacji jak i weryfikacji zastosowane zostały materiały wideo o długości 2000 klatek. Z danych Chokepoint stworzone zostały dwa przypadki testowe (oznaczone odpowiednio literami A i B).

Wartości poszczególnych miar jakości metod OFA_x oraz $FSA_{w_{OPT}, \lambda_{OPT}, A_{OPT}}$ otrzymane w procesie optymalizacji dla każdego przypadku zestawione zostały w lewej stronie tabeli 1. Po wyznaczeniu maksimum, każdy algorytm $FSA_{w_{OPT}, \lambda_{OPT}, A_{OPT}}$ poddany został weryfikacji na innym nagraniu tego samego rodzaju. Wyniki weryfikacji znajdują się z prawej strony tabeli - proporcje błędów każdego rodzaju w procesie weryfikacji wyróżnione zostały pogrubioną czcionką. Zauważyć można dużą poprawę jakości pod względem kryterium IBR. Dla zachowania przejrzystości, pominięte zostały wartości Q_{FSA} , które nie niosą ze sobą więcej informacji dla interpretacji niż ich składowe, które zawarte są w tabeli.

Wyróżnia się również szczególna problematyczność miary MBR, w przypadku której optymalizacja zdecydowanie zawodziła, w najlepszym razie nie pogarszając oryginalnego wyniku. Ponadto, przy większych wartościach parametru L wartość MBR łatwo spadała do zera, co czyniło jej poprawę niemożliwą. Aby zachować możliwość porównywania wyników algorytmów w procesie optymalizacji, wartość MBR w przypadkach oznaczonych symbolem X została pominięta w obliczaniu współczynnika.

Testowany algorytm nie poprawił znacznie wartości wszystkich czterech kryteriów. Jednak ze względu na minimalny koszt jego zastosowania oraz większą regularność wynikowego ciągu klasyfikacji, wprowadzone zmiany w klasyfikacji uznać możemy za wartościowe. Należy zauważyć, że z pomocą metody FSA nie są poprawiane znaczne błędy klasyfikacji, takie jak całkowicie nierozpoznane sceny - więc już z samej konstrukcji metody wynika brak możliwości uzyskania bardzo dużej poprawy.

Znalezione maksima z przeszukiwanej w trakcie optymalizacji przestrzeni parametrów w , λ , A przedstawione zostały w tabeli 2. Przede wszystkim zauważalne są dużo większe szerokości okna dla strumienia z większymi zakłóceniami. Większość wyznaczonych wartości λ znalazła się w górnej połowie zakresu dopuszczalnych wartości, co eksponuje znaczenie również nieco bardziej odległych od środka klatek.

²National ICT Australia Limited, <https://www.nicta.com.au/about-nicta/>

³<http://arma.sourceforge.net/chokepoint/#licence>

Tabela 1

Zestawienie wyników OFA_x oraz $FSA_{w,\lambda,A}$ uzyskanych w kroku optymalizacji oraz weryfikacji.

L	Dane	Miara	Optymalizacja				Weryfikacja			
			FPR	FNR	MBR	IBR	FPR	FNR	MBR	IBR
5	modelowe	OFA_x	0.14	0.11	0.56	0.84	0.11	0.13	0.53	0.80
		$FSA_{w_{OPT},\lambda_{OPT},A_{OPT}}$	0.14	0.07	0.66	0.48	0.10	0.11	0.75	0.68
		$\frac{FSA}{OFA}$	1.02	0.67	1.17	0.57	0.94	0.92	1.41	0.85
	modelowe z zakłóceniami	OFA_x	0.15	0.23	0.47	0.90	0.09	0.16	0.53	0.81
		$FSA_{w_{OPT},\lambda_{OPT},A_{OPT}}$	0.15	0.20	0.53	0.85	0.09	0.14	0.66	0.76
		$\frac{FSA}{OFA}$	0.99	0.88	1.13	0.94	0.99	0.85	1.24	0.93
	ChokepointA	OFA_x	0.25	0.01	0.38	0.87	0.11	0.03	0.07	0.47
		$FSA_{w_{OPT},\lambda_{OPT},A_{OPT}}$	0.24	0.01	0.38	0.58	0.11	0.03	0.12	0.16
		$\frac{FSA}{OFA}$	0.96	1.00	1.00	0.67	0.96	1.03	1.67	0.33
	ChokepointB	OFA_x	0.10	0.06	0.19	0.51	0.07	0.03	0.02	0.48
		$FSA_{w_{OPT},\lambda_{OPT},A_{OPT}}$	0.09	0.05	0.21	0.21	0.06	0.02	0.02	0.02
		$\frac{FSA}{OFA}$	0.91	0.89	1.12	0.42	0.86	0.88	1.00	0.05
10	modelowe	OFA_x	0.14	0.11	0.25	0.72	0.11	0.13	0.22	0.67
		$FSA_{w_{OPT},\lambda_{OPT},A_{OPT}}$	0.14	0.10	0.25	0.66	0.11	0.12	0.41	0.69
		$\frac{FSA}{OFA}$	0.99	0.95	1.00	0.92	1.03	0.97	1.86	1.03
	modelowe z zakłóceniami	OFA_x	0.15	0.23	0.16	0.84	0.09	0.16	0.25	0.70
		$FSA_{w_{OPT},\lambda_{OPT},A_{OPT}}$	0.14	0.23	0.16	0.83	0.09	0.16	0.25	0.70
		$\frac{FSA}{OFA}$	0.98	1.00	1.00	1.00	1.00	0.99	1.00	0.99
	ChokepointA	OFA_x	0.25	0.01	0.25	0.84	0.11	0.03	0.02	0.44
		$FSA_{w_{OPT},\lambda_{OPT},A_{OPT}}$	0.24	0.01	0.25	0.50	0.11	0.03	0.07	0.11
		$\frac{FSA}{OFA}$	0.96	1.00	1.00	0.59	0.96	1.03	3.00	0.25
	ChokepointB	OFA_x	0.10	0.06	0.02	0.41	0.07	0.03	0.00	0.47
		$FSA_{w_{OPT},\lambda_{OPT},A_{OPT}}$	0.09	0.06	0.02	0.02	0.06	0.02	0.02	0.02
		$\frac{FSA}{OFA}$	0.90	0.90	1.00	0.06	0.84	0.88	X	0.05
20	modelowe	OFA_x	0.14	0.11	0.09	0.66	0.11	0.13	0.06	0.60
		$FSA_{w_{OPT},\lambda_{OPT},A_{OPT}}$	0.14	0.10	0.09	0.49	0.10	0.12	0.31	0.44
		$\frac{FSA}{OFA}$	0.97	0.93	1.00	0.75	0.98	0.97	5.00	0.73
	modelowe z zakłóceniami	OFA_x	0.15	0.23	0.06	0.82	0.09	0.16	0.09	0.64
		$FSA_{w_{OPT},\lambda_{OPT},A_{OPT}}$	0.14	0.23	0.06	0.81	0.10	0.16	0.12	0.63
		$\frac{FSA}{OFA}$	0.97	1.00	1.00	0.99	1.02	0.99	1.33	0.98
	ChokepointA	OFA_x	0.25	0.01	0.12	0.82	0.11	0.03	0.00	0.43
		$FSA_{w_{OPT},\lambda_{OPT},A_{OPT}}$	0.25	0.01	0.12	0.78	0.11	0.03	0.00	0.34
		$\frac{FSA}{OFA}$	1.01	1.00	1.00	0.96	0.96	1.03	X	0.79
	ChokepointB	OFA_x	0.10	0.06	0.00	0.40	0.07	0.03	0.00	0.47
		$FSA_{w_{OPT},\lambda_{OPT},A_{OPT}}$	0.10	0.06	0.00	0.40	0.07	0.03	0.00	0.47
		$\frac{FSA}{OFA}$	1.00	1.00	X	1.00	1.00	1.00	X	1.00
50	modelowe	OFA_x	0.14	0.11	0.00	0.62	0.11	0.13	0.06	0.60
		$FSA_{w_{OPT},\lambda_{OPT},A_{OPT}}$	0.14	0.10	0.00	0.59	0.11	0.12	0.12	0.61
		$\frac{FSA}{OFA}$	1.00	0.98	X	0.95	1.01	0.99	2.00	1.01
	modelowe z zakłóceniami	OFA_x	0.15	0.23	0.00	0.81	0.09	0.16	0.09	0.64
		$FSA_{w_{OPT},\lambda_{OPT},A_{OPT}}$	0.14	0.21	0.09	0.76	0.09	0.13	0.22	0.53
		$\frac{FSA}{OFA}$	0.96	0.92	X	0.94	0.98	0.84	2.33	0.82
	ChokepointA	OFA_x	0.25	0.01	0.00	0.79	0.11	0.03	0.00	0.43
		$FSA_{w_{OPT},\lambda_{OPT},A_{OPT}}$	0.25	0.01	0.00	0.79	0.11	0.03	0.00	0.43
		$\frac{FSA}{OFA}$	1.00	1.00	X	1.00	1.00	1.00	X	1.00
	ChokepointB	OFA_x	0.10	0.06	0.00	0.40	0.07	0.03	0.00	0.47
		$FSA_{w_{OPT},\lambda_{OPT},A_{OPT}}$	0.10	0.06	0.00	0.28	0.06	0.03	0.00	0.31
		$\frac{FSA}{OFA}$	0.97	0.97	X	0.69	0.86	0.94	X	0.65

Tabela 2

Zestawienie wyznaczonych parametrów dla różnych wartości L .

Nagranie	L	w	λ	A
modelowe	5	17	0.968	0.553
	10	17	0.874	0.842
	20	7	0.800	0.658
	50	49	0.989	0.974
modelowe z zakłóceniami	5	49	0.947	0.684
	10	41	1.000	0.947
	20	35	0.968	0.921
	50	45	1.000	0.790
ChokepointA	5	9	0.800	0.526
	10	9	0.800	0.526
	20	11	0.800	0.816
	50	9	0.811	0.842
ChokepointB	5	27	0.905	0.500
	10	29	0.874	0.500
	20	7	0.863	0.947
	50	47	0.937	0.605

5. Wnioski

Osiągnięte wyniki wskazują na zasadność stosowania większych rozmiarów okien wraz ze wzrostem poziomu zakłóceń w klasyfikowanych nagraniach. Dodatkowo, algorytm optymalizacji preferował większe wartości parametru λ , zapobiegając zniwelowaniu wpływu bardziej oddalonych klatek na ostateczny wynik klasyfikacji.

Zaproponowany algorytm FSA jest uniwersalny i może być wykorzystywany dla różnych algorytmów OFA zwracających sekwencje binarnych klasyfikacji. Nie zwiększa przy tym złożoności obliczeniowej klasyfikacji w porównaniu do samej metody OFA. Uzyskane wyniki testów dotyczących zaproponowanej metody są obiecujące, ale wymagają dalszych badań:

- ewaluacji dla innych algorytmów OFA dotyczących konkretnych zastosowań, np. endoskopii czy monitoringu,
- modyfikacji rozpatrywanej metody FSA o nowe funkcje wag w oknie bądź zastosowanie bardziej złożonych metod podejmowania decyzji niż ważone głosowanie,
- doboru właściwych metod optymalizacji wartości parametrów, wynikających z wprowadzonych modyfikacji,
- wykorzystania innych niż binarne metod klasyfikacji i opracowania metod poprawiania ich decyzji.

W przypadku ostatniego punktu możliwe jest proste uogólnienie na niewielką liczbę kategorii za pomocą wektorów binarnych⁴. Tego rodzaju reprezentacja pozwala na stosowanie metod dla klasyfikacji binarnych bezpośrednio dla każdej kategorii,

⁴np. przy podziale na trzy kategorie: $[1, 0, 0]$, $[0, 1, 0]$, $[0, 0, 1]$

wymaga jednak rozważenia sposobu wyboru końcowej klasyfikacji spośród kilku potencjalnych kandydatów.

W ramach testów zaproponowanej metody uwidoczniła się również konieczność dokonania oceny przydatności poszczególnych kryteriów. Dla wielu testowanych zestawów parametrów poprawiane były wszystkie kryteria oprócz MBR, którego ocena może zbyt mocno rzutować na ostatecznym wyniku optymalizacji. W zależności od przypadku, zastosowana miara w dużym stopniu wpływała na ocenę jakości klasyfikacji, albo poprzez znaczne zmiany jej wartości albo w związku z małymi wartościami początkowymi (praktycznie bezbłędna klasyfikacja OFA dla większych wartości L), których dalsza poprawa nie ma już sensu. Pozostałe kryteria (FNR, FPR, IBR) dużo precyzyjniej wyrażają jakość otrzymanych decyzji algorytmu klasyfikacji.

LITERATURA

1. Blokus A., Cychnerski J., Brzeski A.: Accelerating video frames classification with metric based scene segmentation. *International Journal of Innovative Research in Computer and Communication Engineering*, 2014, 2(8):5311–5315.
2. Bradski G.: The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.
3. Cao Y., Liu D., Tavanapong W., Wong J., Oh J., De Groen P. C.: Computer-aided detection of diagnostic and therapeutic operations in colonoscopy videos. *IEEE Transactions on Biomedical Engineering*, 2007, 54(7):1268–1279.
4. Cox D., Snell E. J.: *Analysis of Binary Data, Second Edition*. CRC Press, Washington 1989.
5. Froba B., Kublbeck C.: Face Tracking by Means of Continuous Detection. *Computer Vision and Pattern Recognition Workshop*, IEEE Computer Society, Washington 2004, p. 65.
6. Haji-Maghsoudi O., Talebpour A., Soltanian-Zadeh H., Haji-Maghsoudi N.: Automatic organs' detection in WCE. *The 16th CSI International Symposium on Artificial Intelligence and Signal Processing (AISP 2012)*, IEEE, Shiraz 2012, p. 116–121.
7. Kamkar S., Safabakhsh R.: Vehicle detection, counting and classification in various conditions. *IET Intelligent Transport Systems*, Institution of Engineering and Technology, Stevanage 2016, p. 8.
8. Karargyris A., Bourbakis N.: Wireless Capsule Endoscopy and Endoscopic Imaging: A Survey on Various Methodologies Presented. *IEEE Engineering in Medicine and Biology Magazine*, Houghton 2010, 29(1):72–83.
9. Lovell B., Kootsookos P.: Evaluation of HMM training algorithms for letter hand gesture recognition. In *Proceedings of the 3rd IEEE International Symposium on Signal Processing and Information Technology (IEEE Cat. No.03EX795)*, IEEE, 2003, pp 648–651.
10. Marchand P.: Binomial smoothing filter: A way to avoid some pitfalls of least-squares polynomial smoothing. *Review of Scientific Instruments*, Melville 1983, 54(8):1034.

11. Morales J. L., Nocedal J.: Remark on “algorithm 778: L-BFGS-B: Fortran sub-routines for large-scale bound constrained optimization”. *ACM Transactions on Mathematical Software*, 2011, 38(1):1–4
12. Munzer B., Schoeffmann K., Boszormenyi L.: Detection of Circular Content Area in Endoscopic Videos for Efficient Encoding and Improved Content Analysis, Institute of Information Technology, University Klagenfurt, Klagenfurt 2012, pp. 1-20.
13. Munzer B., Schoeffmann K., Boszormenyi L.: Detection of circular content area in endoscopic videos. In *Proceedings of the 26th IEEE International Symposium on Computer-Based Medical Systems*, IEEE, Porto 2013, p. 534–536.
14. Nash S. G.: Newton-Type Minimization via the Lanczos Method. *SIAM Journal on Numerical Analysis*, Philadelphia 1984, 21(4):770–788.
15. Ojala T., Pietikainen M., Maenpaa T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE 2002 24(7):971–987.
16. Oliphant T. E.: SciPy: Open source scientific tools for Python. *Computing in Science and Engineering*, IEEE, New York 2007, 9:10–20.
17. Lienhart R., Kuranov A., Pisarevsky V.: Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection. *Pattern Recognition: Lecture Notes in Computer Science*, Magdeburg 2003 pp 297-304
18. Ramona M., Richard G., David B.. Vocal detection in music with support vector machines. In *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, IEEE, Las Vegas 2008, p. 1885–1888.
19. Rublee E., Rabaud V., Konolige K., Bradski G.: ORB: An efficient alternative to SIFT or SURF. In *2011 International Conference on Computer Vision*, IEEE, Barcelona 2011., p. 2564–2571.
20. Turaga P.: *Statistical and Geometric Modeling of Spatio-Temporal Patterns for Video Understanding*. PhD thesis, University of Maryland, 2009.
21. Viola P., Jones M.: Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, pages I–511–I–518. IEEE Comput. Soc, 2001.
22. Wong Y., Chen S., Mau S., Sanderson C., Lovell B. C.: Patch-based probabilistic image quality assessment for face selection and improved video-based face recognition. In *CVPR 2011 WORKSHOPS*, IEEE, Colorado Springs 2011, p. 74–81.