



DOI: 10.5604/20830157.1148054

WYKORZYSTANIE GPU DO OKREŚLENIA OBSZARU PŁOMIENIA W WIZYJNYM SYSTEMIE DIAGNOSTYCZNYM

Daniel Sawicki

Politechnika Lubelska, Instytut Elektroniki i Technik Informacyjnych, Zakład Diagnostyki i Analizy Pomiarów

Streszczenie. W pracy przedstawiono możliwości zastosowania jednostki GPU do wyznaczenia krawędzi dla obrazów spalania pyłu węglowego. Porównano czas wykonywania operacji oraz dokładność detekcji brzegu obszaru płomienia. Wykorzystano metodę opartą o model konturu aktywnego Chan-Vese. Wyniki badań pokazują, że nastąpiło znaczne przyspieszenie wykonywania operacji (ok. 400%).

Słowa kluczowe: wykrywanie brzegu obszaru, płomień, spalanie, GPU

ВИКОРИСТАННЯ GPU ДЛЯ ВИЗНАЧЕННЯ ОБЛАСТІ ПОЛУМ'Я У ВІЗУАЛЬНІЙ ДІАГНОСТИЧНІЙ СИСТЕМІ GPU

Анотація. У статті представлені можливості використання пристрою GPU для визначення країв зображення при спалюванні вугільного пилу. Порівняно час виконання і точність виявлення краю області полум'я. Використовується метод, заснований на моделі активного контуру Chan-Vese. Результати досліджень показують, що відбулося істотне прискорення операцій (прибл. 400%).

Ключові слова: виявлення краю області, полум'я, спалення, GPU

USING THE GPU TO DETERMINING THE AREA THE FLAME IN THE VISION DIAGNOSTIC SYSTEM

Abstract. This paper presents possibility of using GPU to determine the edge of the image pulverized combustion coal. Compared are the operation time and the accuracy of edge detection area of the flame. Used a method based on the active contour model Chan-Vese. The results show that there was a significant acceleration of the operation (up to 400%).

Keywords: edge detection, flame, combustion, GPU

Wstęp

Wykorzystanie płomienia jako źródła informacji o procesie spalania jest jednym ze sposobów diagnozowania tego procesu. Dzięki analizie obrazu płomienia można uzyskać informację o stanie procesu praktycznie bez żadnych opóźnień. Jest to szczególnie istotne w przypadku spalania paliw charakteryzujących się dużą zmiennością właściwości fizyko-chemicznych. Do tej grupy należą m.in. mieszaniny biomasy i węgla, których spalanie jest najbardziej rozpowszechnionym w Polsce sposobem wykorzystania paliw odnawialnych [5, 6, 7].

Płomień towarzyszy egzotermicznym reakcjom utleniania. Obecność płomienia związana jest więc z miejscem w przestrzeni, gdzie taki rodzaj reakcji występuje. Trudno jest jednak w takim przypadku wskazać granicę oddzielającą przestrzeń, w której spalanie występuje, ponieważ stężenia reagentów nie zmieniają się w sposób skokowy. Głównym źródłem promieniowania w płomieniu pyłowym jest obłok rozgrzanych do wysokiej temperatury cząstek stałych (pyłu węglowego, sadzy, popiołu itp.). Ponieważ koncentracja świecących cząstek nie zmienia się skokowo, nie można wyróżnić krawędzi płomienia, jak np. w przypadku ciała stałego. Występujące w płomieniu zjawiska pochłaniania i rozpraszania promieniowania wprowadzają dodatkowe niejednoznaczności w określeniu granicy płomienia. Ważnym aspektem jest także czas wykonywania operacji wykrywania krawędzi ponieważ od tego uzależniona jest możliwość sterowania procesem spalania. Z tego względu wykorzystano metodę aktywnego konturu [1-3] dzięki której najdokładniej wyznaczono brzeg płomienia. [7] oraz jednostkę GPU w celu przyspieszenia obliczeń [4]. Badania wykonano dla obrazów zarejestrowanych w czasie testów spalania mieszanin pyłu węglowego i biomasy (trawa) w warunkach laboratoryjnych przy różnych wydatkach powietrza i paliwa.

Вступ

Використання полум'я в якості джерела інформації про процес згоряння є одним із способів діагностики цього процесу. Аналізуючи образ полум'я можна отримати інформацію про процес практично без будь-яких затримок. Це особливо важливо у випадку спалювання палива, яке характеризується високою змінністю фізико-хімічних властивостей. До цієї групи належать суміші біомаси та вугілля, спалювання яких є найбільш поширеним в Польщі способом використання поновлюваних видів палива [5, 6, 7].

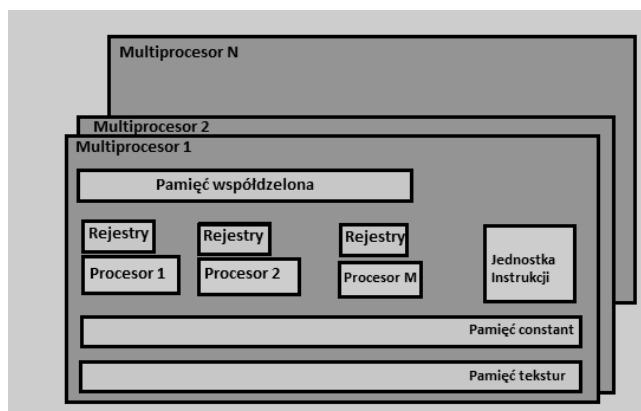
Полум'я товарищує екзотермічним реакціям окислення. Наявність полум'я, пов'язано з місцем у просторі, де є цей тип реакції. Важко в цьому випадку вказати межу, яка розділяє простір, в якому відбувається згоряння, оскільки концентрація реагентів не змінюється в різкий спосіб. Основним джерелом випромінювання в пиловому полум'ї є хмара розігрітих до високої температури твердих частинок (вугільного пилу, сажі, попелу і т.д.). Оскільки концентрація люмінесцируючих частинок не змінюється різко, не можна відрізнити країв полум'я, наприклад, для твердої речовини. Явища, що відбуваються в полум'ї, поглинання і розсіяння випромінювання вводять додаткову неоднозначність у визначенні меж полум'я. Важливим аспектом є також час виявлення краю, так як це залежить від здатності контролювати процес горіння. Таким чином, був використаний метод активного контуру [1-3] завдяки якому було точно визначено краї полум'я [7] і GPU для прискорення обчислень [4]. Випробовування проводилися для зображень, записаних під час випробувань згоряння суміші пилоподібного вугілля та біомаси (трава) в лабораторних умовах з різними витратами повітря і палива.

1. GPU

W budowie procesorów komputerowych, wielordzeniowość jest powszechnie stosowana, lecz w budowę procesorów graficznych, była stosowana dużo wcześniej. Przyczyną jest łatwość zrównoleglenia algorytmów używanych w grafice. Wykorzystanie wielordzeniowych kart graficznych, tzw. Graphics Processing Unit (GPU) do dowolnych obliczeń numerycznych znacznie rozszerzyło ich możliwości. Najbardziej popularną technologią obliczeń na GPU jest CUDA (Compute Unified Device Architecture) której twórcą jest firma NVIDIA [4, 8, 9].

Zadania do których wykorzystywane są procesory graficzne znacząco wpłynęły na różnice w budowie CPU i GPU. GPU zawiera wiele jednostek arytmetyczno-logicznych ALU oraz jednostek zmienoprzecinkowych FPU. Jest on wykorzystywany do rozwiązywania zadań, które mogą być łatwo zrównoległone (te same operacje są wykonywane na dużej ilości danych). Wprowadzono pojęcie architektury SIMT (Single Instruction Multiple Threads) – jedna instrukcja wiele wątków. Grupą wątków można zarządzać za pomocą pojedynczych instrukcji kodu źródłowego. W SIMT wątki posiadają własne liczniki instrukcji i mogą się rozgałęziać za pomocą instrukcji warunkowych pomimo wspólnego adresu początkowego podprogramu. Występuje także bardzo duża liczba wątków, oraz minimalizacja liczby operacji wykonywanych przez jeden wątek, a zwłaszcza operacji dostępu do pamięci. Z powodu różnego zapotrzebowania na moc obliczeniową dla różnych typów przetwarzanych obiektów — wierzchołków lub fragmentów przez różne aplikacje utworzono architekturę, w której istnieją uniwersalne jednostki przetwarzania. Nazywane są procesorami strumieniowymi lub w przypadku omawianej technologii - rdzeniami CUDA [8, 9].

Karta graficzna zawiera od kilku do kilkudziesięciu tzw. multiprocessorów, pamięć globalną oraz niewielką pamięć constant (rysunek 1). W zależności od konkretnego modelu karty graficznej liczba multiprocessorów jest różna. Różnice są znaczne, dla przykładu na kartach graficznych przeznaczonych dla komputerów przenośnych, mogą się znajdować 4 multiprocessory. Natomiast egzemplarze kart graficznych zaliczane do klasy o wysokiej wydajności, mogą mieć 16 lub więcej multiprocessorów.



Rys. 1. Układ procesorów w karcie graficznej [8]

Rdzeń zawiera 8 procesorów skalarnych, różne rodzaje pamięci ze specyficznym przeznaczeniem. Pamięć globalna jest dostępna dla wszystkich wątków. Jej przepustowość wynosi do 100 GB/s, ale opóźnienie przy dostępie wynosi kilkaset cykli zegara, więc zaleca się minimalizowanie używania tej pamięci w programach. Pamięć constant jest zapisywana przed uruchomieniem wszystkich operacji na GPU, zapis do niej nie jest później możliwy. Wykorzystywana jest do przechowywania często używanych danych, które nie są modyfikowane w trakcie działania operacji na GPU. Pamięć lokalna znajduje się w pamięci globalnej, przechowywane są w niej parametry wywołania funkcji GPU lub zmienne lokalne. Pamięć dzielona jest używana jako cache pamięci globalnej lub do komunikacji między wątkami w danym multiprocessorze [8, 9].

1. GPU

При побудові комп'ютерних процесорів, багатоядерність широко використовується, але в конструкції графічних процесорів використовували набагато раніше. Причина в легкості розпаралелювання алгоритмів, використовуваних у графіці. Використання багатоядерних графічних карт, Graphics Processing Unit (GPU) для довільних цифрових обрахунків значно розширило їх можливості. Найбільш популярна технологія обрахунків GPU є CUDA (Compute Unified Device Architecture), засновником якої є фірма NVIDIA [4, 8, 9].

Завдання, для яких використовуються графічні процесори, значно вплинули на відмінності в будівництві CPU і GPU. GPU містить багато арифметичних і логічних пристроїв ALU і пристроїв з плаваючою точкою одиниць FPU. Він використовується, щоб вирішити завдання, які можуть бути легко розпаралелені (ті ж операції виконуються на великих обсягах даних). Введено поняття архітектури SIMT (Single Instruction Multiple Threads) - одна інструкція кілька джерел. Групою тем, можна керувати за допомогою одиночних інструкцій вихідного коду. В SIMT жерела мають свої власні лічильники інструкції і можуть розгалужуватися з використанням умовних операторів, незважаючи на спільну адресу початкової підпрограми. Існує також велика кількість тем, а також мінімізація кількості операцій, які виконуються однією темою, а особливо операцій доступу до пам'яті. Через різну потребу потужності при обрахунках, які необхідні для різних типів оброблюваних об'єктів - вершин або фрагментів, створених різними додатками, в яких є універсальний блок обробки. Вони називаються потоковими процесорами, або у разі обговорюваної технології - ядрами CUDA [8, 9].

Відеокарта містить від кількох до десятків так званих мультіпроцесорів, глобальну пам'ять і невелику сталу пам'ять (малюнок 1). В залежності від конкретної моделі графічної карти кількість мультіпроцесорів відрізняється. Відмінності істотні, наприклад, для графічних карт, призначених для портативних комп'ютерів, може бути 4 мультіпроцесори. З іншого боку, копії відеокарт, включених в клас високопродуктивних, може мати 16 або більше мультіпроцесорів.

Мал. 1. Процесорна система у відеокарті [8]

Ядро містить 8 скалярних процесорів, різні види пам'яті з конкретним призначенням. Глобальна пам'ять доступна для всіх тем. Її потужність становить до 100 Гб/с, але затримка при доступі складає сотні циклів годинника, тому бажано, звести до мінімуму використання цієї пам'яті в програмах. Постійна пам'ять записується перед виконанням всіх операцій на GPU, дописування до неї є неможливе. Використовується для зберігання часто використовуваних даних, які не змінюються при операціях виконання на GPU. Локальна пам'ять знаходиться в глобальній пам'яті зберігаються в ній параметри виклику функції GPU або локальних змінних. Спільна пам'ять використовується як кеш-пам'яті або глобальної комунікації між темами в мультіпроцесорі [8, 9].

1.1. Architektura CUDA

W tej architekturze istnieje szereg charakterystycznych cech m. in. brak bezpośredniego dostępu do sprzętu (brak udostępnionego assemblera), przesyłanie kodu maszynowego za pośrednictwem sterownika. Istotą przesyłania kodu maszynowego za pomocą sterownika jest możliwość jego translacji dla docelowego urządzenia. Pozwala to na wprowadzenie ulepszeń optymalizacyjnych po stronie nowego sprzętu, w wyniku których nie jest konieczna ponowna ingerencja w kod przeznaczony dla GPU. Architektura CUDA została tak zaprojektowana, aby nie było wielu ograniczeń, które by uniemożliwiały wykorzystanie wcześniejszych procesorów graficznych do ogólnych zastosowań. W odróżnieniu od poprzednich generacji procesorów, w których jednostki wykonujące obliczenia były podzielone na specjalne funkcje (tzw. shadery) wierzchołków i pikseli, w architekturze CUDA zastosowano jeden połączony potok przetwarzania. Dzięki temu program wykonujący obliczenia ma do dyspozycji wszystkie ALU procesora. Dodatkowo jednostkom wykonawczym GPU zezwolono na swobodny dostęp do pamięci w celu odczytu i zapisu, a także do zarządzanej programowo pamięci podręcznej zwanej pamięcią wspólną. Wykorzystując tę architekturę osiągnięto zwiększenie wydajności programów nawet o kilka rzędów wielkości w porównaniu z wcześniejszymi implementacjami. Z tego względu podjęto próbę wykorzystania technologii GPU do wyznaczenia brzegu obszaru płomienia algorytmem wymagającym dużej szybkości obliczeń. [8, 9].

2. Model konturu aktywnego Chan-Vese

Algorytm Chan-Vese do segmentacji obrazu wykorzystuje uproszczony model Mumford-Shah i funkcję zbiorów poziomicowych [1-3]. Algorytm ten nie wymaga stosowania wielu obszarów startowych w celu wykrycia bardziej złożonych kształtów obiektów. W tej metodzie Ω jest obszarem obrazu $I(x, y)$, który można podzielić na wewnętrzny Ω_1 i zewnętrzny Ω_2 obszar ograniczony konturem ϕ o intensywności odpowiednio c_1 i c_2 . Funkcja celu konturu aktywnego została zdefiniowana następująco [3]:

$$F(\phi, c_1, c_2) = \mu \cdot L(\phi) + \nu \cdot S_0(\phi) + \lambda_1 \int_{\Omega_1} |I - c_1|^2 dx dy + \lambda_2 \int_{\Omega_2} |I - c_2|^2 dx dy \quad (1)$$

gdzie: $L(\phi)$ jest długością ϕ , $S_0(\phi)$ jest wewnętrznym obszarem ϕ , $\mu, \nu \geq 0$, $\lambda_1, \lambda_2 > 0$ są współczynnikami odpowiednich wyrażeń funkcji celu. Minimalizując funkcję celu (11) otrzymano następujące równania [3]:

$$\begin{cases} c_1(\phi) = \frac{\int_{\Omega} I(x, y) H_\varepsilon(\phi) dx dy}{\int_{\Omega} H_\varepsilon(\phi) dx dy} \\ c_2(\phi) = \frac{\int_{\Omega} I(x, y) (1 - H_\varepsilon(\phi)) dx dy}{\int_{\Omega} (1 - H_\varepsilon(\phi)) dx dy} \\ \frac{\partial \phi}{\partial t} = \delta_\varepsilon(\phi) \left[\mu \nabla \cdot \frac{\nabla \phi}{|\nabla \phi|} - \nu - \lambda_1 (I(x, y) - c_1)^2 \right. \\ \left. + \lambda_2 (I(x, y) - c_2)^2 \right] \\ \phi(x, y, t = 0) = \phi_0(x, y) \end{cases} \quad (2)$$

gdzie: Ω jest obszarem ϕ , funkcja skokowa i impulsowa zostały zdefiniowane następująco:

$$\begin{aligned} H(z) &= \begin{cases} 1, z \geq 0 \\ 0, z < 0 \end{cases} \\ \delta(z) &= \frac{dH(z)}{dz} \end{aligned} \quad (3)$$

1.1. Архітектура CUDA

В цій архітектурі є характеристичні особливості, наприклад, відсутність прямого доступу до пристрою (відсутність доступу до асемблера), передача машинного коду через контролер. Суть передачі машинного коду, за допомогою контролера є можливість його передачі для цільового пристрою. Це дозволяє впровадити оптимальні поліпшення по стороні нової техніки, в результаті якого немає необхідності повторно втручання в код призначений для GPU. Архітектура CUDA була розроблена таким чином, щоб не було багато обмежень, які перешкоджали б використанню попередніх графічних процесів для загального застосування. На відміну від попереднього покоління процесорів, в яких підрозділи, що ведуть розрахунки були розділені на спеціальні функції (тобто шейдери), вершин і пікселів в архітектурі CUDA використовує один поєднаний потік обробки. Таким чином, програма, яка виконує розрахунок, має всі одиниці арифметико-логічного процесора. Додатково виконуючі одиниці GPU мають вільний доступ до пам'яті для читання і запису, а також до підручної пам'яті керованої через програму, яка називається спільною пам'яттю. Використовуючи цю архітектуру продуктивність програм була збільшена на кілька порядків у порівнянні з попередніми реалізаціями. З цієї причини, була зроблена спроба використовувати GPU технології, щоб визначити межу полум'я алгоритмом, який вимагає швидких розрахунків [8, 9].

2. Модель активного контуру Chan-Vese

Алгоритм Chan-Vese для сегментації зображень використовує спрощену модель Mumford-Shah і функцію заданого рівня [1-3]. Цей алгоритм не вимагає використання декількох областей старту, щоб виявити більш складні форми об'єктів. У цьому методі, Ω є область зображення $I(X, Y)$, які можуть бути розділені на внутрішні Ω_1 і зовнішні Ω_2 область обмежена контуром ϕ з інтенсивністю c_1 і c_2 . Цільова функція активного контура визначається наступним чином [3]:

де: $L(\phi)$ є довжиною ϕ , $S_0(\phi)$ є внутрішньою областю ϕ , $\mu, \nu \geq 0$, $\lambda_1, \lambda_2 > 0$ коефіцієнти відповідних виразів цільової функції. Мінімізація цільової функції (11) отримано наступне рівняння [3]:

W obliczeniach wykorzystano następujące równania funkcji skokowej i funkcji celu [3]:

$$H_\varepsilon(z) = \frac{1}{2} \left(1 + \frac{2}{\pi} \arctg\left(\frac{z}{\varepsilon}\right) \right) \quad (4)$$

$$\delta_\varepsilon(z) = \frac{1}{\pi} \frac{\varepsilon}{\varepsilon^2 + z^2} \quad (5)$$

$$F(x, y) = -v - \lambda_1 (I(x, y) - c_1)^2 + \lambda_2 (I(x, y) - c_2)^2 \quad (6)$$

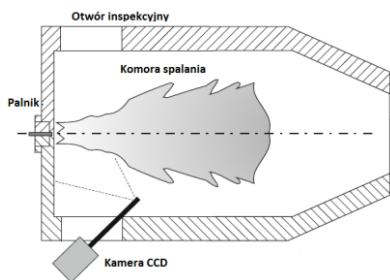
gdzie ε jest wartością stałą.

де ε має сталі значення.

3. Laboratoryjne testy spalania

Testy spalania pyłu węglowego i biomasy zostały przeprowadzone na stanowisku laboratoryjnym, którego zasadniczym elementem jest cylindryczna komora spalania o długości 2,5 m i średnicy ok. 0,7 m. Wewnątrz komory znajduje się wykonany w skali 1:10 model wirowego palnika pyłowego oraz palnik olejowy. Obydwa palniki umieszczone są w jednej osi i mogą pracować jednocześnie. Mieszanka paliwowa wraz z powietrzem pierwotnym dostarczana jest do palnika pyłoprzewodem. Regulację wydatku węgla i biomasy dokonywana jest poprzez zmianę prędkości obrotowej jego podajnika. Na przebieg procesu spalania istotny wpływ ma ponadto wartość przepływu powietrza wtórnego.

Komora spalania zaopatrzona jest w dwa podłużne otwory rozmieszczone po obu jej stronach, które umożliwiają obserwację płomienia. Widok komory spalania wraz z zaznaczeniem umiejscowieniem kamery został przedstawiony na rysunku 2.



Rys. 2. Widok ogólny stanowiska pomiarowego
Мал. 2. Загальний вигляд місця вимірювань

3.1. Badanie procesu spalania

Badania procesu spalania wykonano dla czterech wariantów z różnymi ustawieniami mocy cieplnej (P_{th}), współczynnika wydatku powietrza (λ) dla stałej zawartości biomasy (20%). Podczas testu spalania, obrazy płomienia były rejestrowane przez system rejestracji w skład którego wchodziła kamera CCD współpracująca z obrazowodem oraz pamięć masowa. Obrazy płomienia w skali 24-bitowej RGB ograniczone do rozmiarów 1280x1024 pikseli, rejestrowane były szybkością 150 klatek/s (bez kompresji) i poddawane późniejszej analizie. Dokładne wartości mocy cieplnej i współczynnika wydatku powietrza zestawiono w tabeli 1.

4. Wyniki

Na rysunkach 3-10 przedstawiono wyniki przeprowadzonej segmentacji dla wybranych obrazów płomienia z wykorzystaniem metody konturu aktywnego dla CPU i GPU. W modelu konturu aktywnego użyto następujących parametrów: $\lambda_1 = \lambda_2 = 1$, $\varepsilon = 1$. Przedstawiono wyniki wybranych obrazów płomienia zarejestrowanych przy współspalania pyłu węglowego i biomasy. Testowane obrazy obejmowały strefę w pobliżu wylotu palnika pod kątem ok. 45° do osi palnika.

3. Лабораторні тести згоряння

Тести спалювання пилоподібного вугілля та біомаси проводили в лабораторії, де істотним елементом є циліндрична камера згоряння з довжиною 2,5 м і діаметром приблизно 0,7 м. В середині камери знаходиться виконана в масштабі 1:10 модель вихрового пальника і масляного пальника. Обидва пальники розташовані на одній осі та можуть працювати одночасно. Суміш палива з первинним повітрям подається до пальника через пилопровід. Використання вугілля та біомаси регулюється шляхом зміни швидкості подачі. У процесі горіння також робить істотний вплив швидкість вторинного повітряного потоку.

Камера згоряння оснащена двома отворами у вигляді пази, розташованих з обох сторін, що дозволяє спостереження за полум'ям. Вигляд камери згоряння разом з обраним розташуванням камери показаний на мал.2.

Tabela 1. Warianty procesu spalania
Таблиця 1. Варіанти процесу згоряння

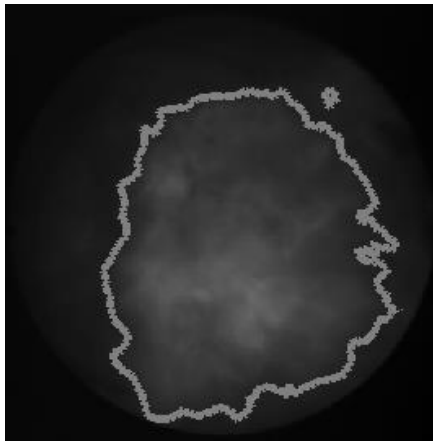
Wariant Варіант	P_{th} (kW)	λ
1	400	0.65
2	300	0.65
3	300	0.75
4	250	0.65

3.1. Дослідження процесу горіння

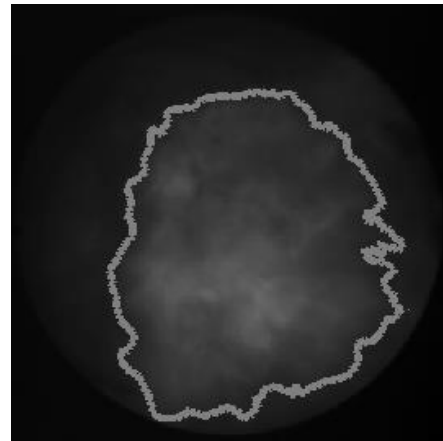
Дослідження процесу згоряння виконано для чотирьох варіантів, з різними настройками теплової енергії (P_{th}), швидкість потоку повітря (λ) для сталого вмісту біомаси (20%). Під час спалювання, зображення полум'я було зареєстроване в системі реєстрації, яка включала CCD камеру, яка співпрацювала з введенням образу та збереженням. Зображення полум'я в 24-бітній шкалі RGB обмежується до розмірів 1280x1024 пікселів зображення полум'я були записані зі швидкістю 150 кліток/с (без стиснення) і піддані подальшому аналізу. Точне значення коефіцієнта теплової енергії та витрати повітря наведені в таблиці 1.

4. Результати

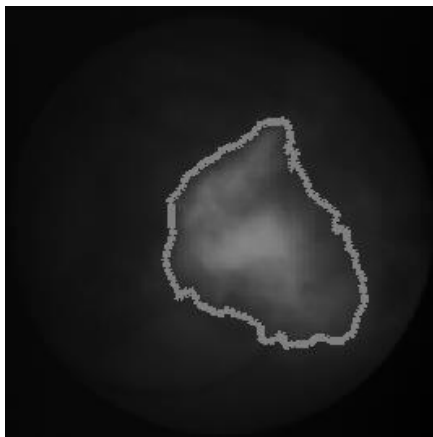
На малюнках 3-10 показані результати сегментації для вибраних зображень полум'я за допомогою методу активного контуру для CPU і GPU. В моделі активного контуру використовуються наступні параметри: $\lambda_1 = \lambda_2 = 1$, $\varepsilon = 1$. Представлені результати вибраних зображень полум'ям записані при спільному спалюванні пилоподібного вугілля та біомаси. Тестові зображення включають в себе область поблизу випускного отвору пальника під кутом прибіл. 45° до осі пальника.



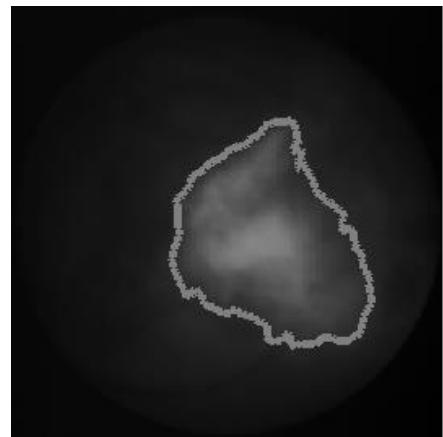
Rys. 3. Wyznaczony brzeg obszaru płomienia dla wariantu 1 z wykorzystaniem CPU



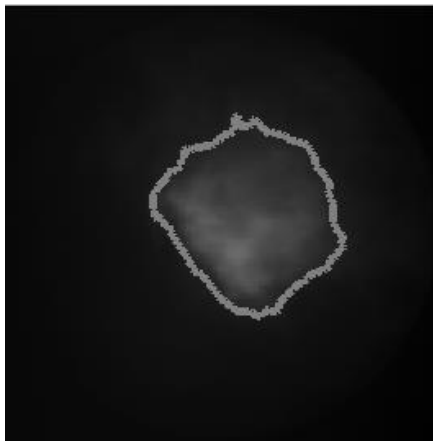
Rys. 4. Wyznaczony brzeg obszaru płomienia dla wariantu 1 z wykorzystaniem GPU



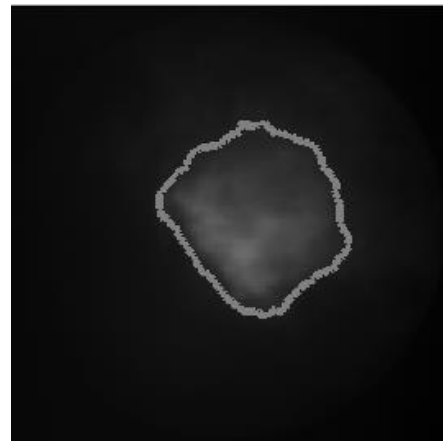
Rys. 5. Wyznaczony brzeg obszaru płomienia dla wariantu 1 z wykorzystaniem CPU



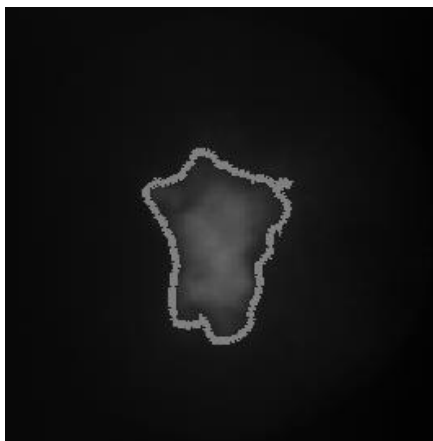
Rys. 6. Wyznaczony brzeg obszaru płomienia dla wariantu 1 z wykorzystaniem GPU



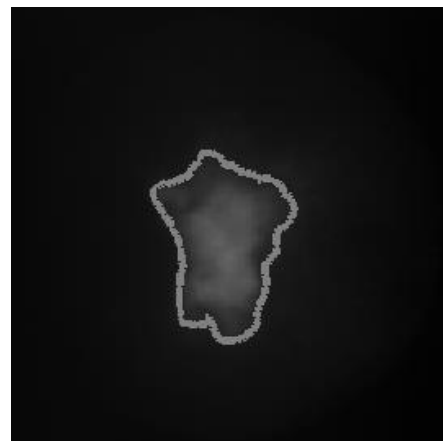
Rys. 7. Wyznaczony brzeg obszaru płomienia dla wariantu 1 z wykorzystaniem CPU



Rys. 8. Wyznaczony brzeg obszaru płomienia dla wariantu 1 z wykorzystaniem GPU



Rys. 9. Wyznaczony brzeg obszaru płomienia dla wariantu 1 z wykorzystaniem CPU



Rys. 10. Wyznaczony brzeg obszaru płomienia dla wariantu 1 z wykorzystaniem GPU

Wyznaczono czas wykonywania operacji wykrywania brzegu obszaru z wykorzystaniem procesora CPU i GPU. Wyniki przedstawiono w tabeli 1.

Tabela 2. Porównanie czasu wykonywania operacji

Wariant Варіант	CPU [s]	GPU [s]	Wzrost [%] Зростання [%]
1	0.158377	0.0362538	436.856
2	0.155893	0.0360772	432.108
3	0.159715	0.0375965	424.814
4	0.15579	0.0357528	435.741

Podsumowanie

Z przedstawionych obrazów testowych można wyciągnąć następujące wnioski. Zastosowanie przedstawionych metod wykrywania krawędzi dało podobne wyniki odnośnie dokładności wykrycia krawędzi. Kontur uzyskany za pomocą metody Chan-Vese z wykorzystaniem procesorów CPU i GPU jest zamknięty. Pozwala to na analizę kształtu płomienia z wykorzystaniem geometrycznych wskaźników kształtu, deskryptorów Fouriera lub falkowych. Dzięki wykorzystaniu procesora GPU możliwa będzie ocena stanu procesu spalania na podstawie obrazów płomienia rejestrowanych w czasie rzeczywistym. Biorąc powyższe pod uwagę, istotny jest czas w którym następuje wyznaczenie krawędzi ze względu na możliwość pomijania klatek w przetwarzanej sekwencji obrazów (przyspieszenie ok 400%). Dalsze prace skoncentrowane będą na dokonaniu klasyfikacji stanu procesu spalania z wykorzystaniem metod statystycznych.

Podziękowania

Praca powstała w ramach projektu *PL-NTU Transgraniczna wymiana doświadczeń* PBU.03.01.00-06-386/11-00 współfinansowanego w ramach Programu Współpracy Transgranicznej Polska-Białoruś-Ukraina 2007-2013 finansowanego ze środków Unii Europejskiej w ramach Europejskiego Instrumentu Sąsiedztwa i Partnerstwa.

Niniejsza publikacja została stworzona przy pomocy Unii Europejskiej. Wyłączną odpowiedzialność za zawartość niniejszej publikacji ponosi Daniel Sawicki oraz w żaden sposób nie może być ona postrzegana jako odzwierciedlenie poglądów Unii Europejskiej.

Literatura || Література

- [1] Canny, J.: A Computational Approach To Edge Detection, IEEE Trans. Pattern Analysis and Machine Intelligence, 8, 1986, 679-714.
- [2] Chan T., Sandberg Y. L.: Active contours without edges for Vector-valued Image, Journal of Visual Communication and Image Representation 11, 2000, 130-141.
- [3] Chan T., Vese L.: Active contours without edges, IEEE Transaction on Image Processing 10, 2001, 266-277.
- [4] Fung J., Mann S.: Using graphics devices in reverse: GPU-based Image Processing and Computer Vision, IEEE International Conference on Multimedia and Expo, Hannover, Germany, 2008, 9-12.
- [5] Golec T.: Współspalanie biomasy w kotłach energetycznych, Energetyka i Ekologia 7, 2004, 437-444.

Mgr inż. Daniel Sawicki
e-mail: d.sawicki@pollub.pl

Ukończył studia o kierunku elektrotechnika na Wydziale Elektrotechniki i Informatyki Politechniki Lubelskiej w 2006r. Obecnie jest zatrudniony jako asystent w Instytucie Elektroniki i Techniki Informacyjnych Politechniki Lubelskiej w Zakładzie Diagnostyki i Analizy pomiarów. Jego prace badawcze związane są ze zwiększeniem efektywności współspalania węgla kamiennego i biomasy z wykorzystaniem analizy obrazu.



Магістр інж. Даниєль Савіцкі
e-mail: d.sawicki@pollub.pl

Закінчив напрямок електротехніка на факультеті електротехніки та інформатики в Люблінській Політехніці в 2006 році. В даний час працює помічником в Інституті електротехніки та інформатики в Люблінській Політехніці на кафедрі діагностики та аналізу вимірювань. Його дослідницькі роботи пов'язані зі збільшенням ефективності спалювання вугілля та біомаси з використанням аналізу зображення.

Визначено час процесу виявлення краю за допомогою центрального і графічного процесорів. Отримані результати наведені в таблиці 1.

Таблиця 2. Порівняння часу виконання операції

Підсумок

З представлених тестових зображень, можна зробити наступні висновки. Використання запропонованих методів виявлення краю дали подібні результати, з точки зору точності виявлення краю. Контур, отриманий методом Chan-Vese з використання процесорів CPU і GPU є закритий. Це дозволяє проводити аналіз форми полум'я використовуючи геометричні показники форми, дескрипторів Фур'є або вейвлет. При використанні GPU можна буде оцінити стан процесу згоряння на підставі зображення полум'я, записаного в режимі реального часу. З урахуванням вищесказаного, важливим є час, в якому відбувається визначення країв через здатність пропускати клітки в обробленій послідовності зображень (прискорення приблизно 400%). Подальша робота буде зосереджена на класифікації стану процесу згоряння, використовуючи статистичні методи.

Завдяки

Робота створена в рамках проекту *PL-NTU Прикордонного обміну досвідом* PBU.03.01.00-06-386/11-00, фінансованих в рамках Програми Транскордонного Співробітництва Польща-Білорусь-Україна 2007-2013 фінансується Європейським Союзом в рамках Європейського Інструменту Суспільства та Партнерства.

Ця публікація була створена за допомогою Європейського Союзу. Відповідальність за зміст цієї публікації лежить на Даниєль Савіцкі, і жодним чином не може розглядатися як відображення поглядів Європейського Союзу.