

Modele dyfuzyjne w zadaniu super-rozdzielczości wideo na konsumenckich kartach graficznych

Daniel Machniak

promotor: prof. dr hab. inż. Przemysław Rokita

Instytut Informatyki

2026-01-20

Plan prezentacji

1. Cel pracy
2. Wprowadzenie do problematyki VSR
3. Podstawy teoretyczne: Transformery i dyfuzja
4. Analiza architektury FlashVSR
5. Optymalizacja i implementacja
6. Ewaluacja i podsumowanie

Cel pracy

Celem pracy jest opracowanie potoku przetwarzania w zadaniu super-rozdzielczości wideo (VSR) z wykorzystaniem modeli dyfuzyjnych na kartach graficznych klasy konsumenckiej. Istotnym elementem pracy jest także zbadanie wpływu technik optymalizacji na jakość rekonstruowanych materiałów.

Wprowadzenie do problematyki VSR

Sformułowanie zadania super-rozdzielczości wideo...

Zadanie super-rozdzielczości wideo (ang. *video super-resolution*, **VSR**) to proces rekonstrukcji sekwencji wideo o wysokiej rozdzielczości (HR) na podstawie materiału wejściowego o niskiej rozdzielczości (LR), wykorzystujący przestrzenno-czasowe zależności między sąsiednimi klatkami do odzyskania brakujących szczegółów [1].

Sformułowanie zadania super-rozdzielczości wideo...

Proces degradacji klatek HR można formalnie zapisać jako:

$$I_i = \phi\left(\hat{I}_i, \left\{\hat{I}_j\right\}_{j=i-N}^{i+N}; \theta_\alpha\right), \quad (1)$$

- I - klatki LR,
- \hat{I} - klatki HR,
- ϕ - funkcja degradacji,
- θ_α - czynniki degradacji (szum, rozmycie, kompresja).

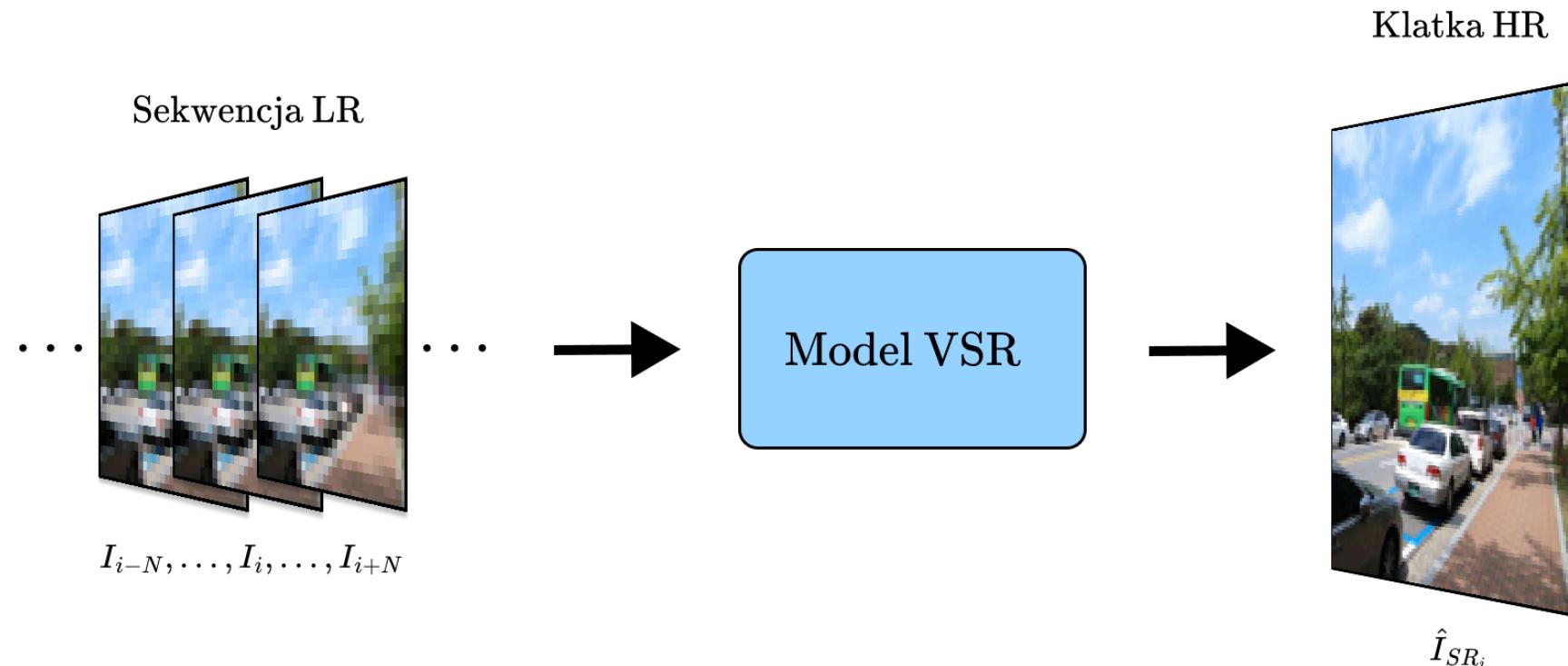
Sformułowanie zadania super-rozdzielczości wideo...

VSR można zatem zdefiniować jako proces odwrotny równania (1) i opisać wzorem:

$$\hat{I}_{\text{SR}_i} = f_{\text{VSR}} \left(I_i, \left\{ I_j \right\}_{j=i-N}^{i+N}; \theta_{f_{\text{VSR}}} \right), \quad (2)$$

- \hat{I}_{SR} - zrekonstruowane klatki HR,
- I - klatki LR,
- f_{VSR} - model super-rozdzielczości,
- $\theta_{f_{\text{VSR}}}$ - uczone parametry modelu.

Sformułowanie zadania super-rozdzielczości wideo



Rysunek 1: Schemat zadania super-rozdzielczości wideo (VSR).

Kluczowe wyzwania w VSR

1. **Rekonstrukcja szczegółów:** Zadaniem modelu jest odtworzenie realistycznych tekstur i krawędzi, które zostały bezpowrotnie utracone w procesie obniżania rozdzielczości.
2. **Wykorzystanie zależności czasowych:** Algorytm musi efektywnie pobierać brakujące informacje z klatek sąsiednich.
3. **Zapewnienie spójności wideo:** Kluczowym wymogiem jest zachowanie stabilności obrazu w czasie, aby uniknąć nienaturalnego migotania pomiędzy kolejnymi klatkami sekwencji.

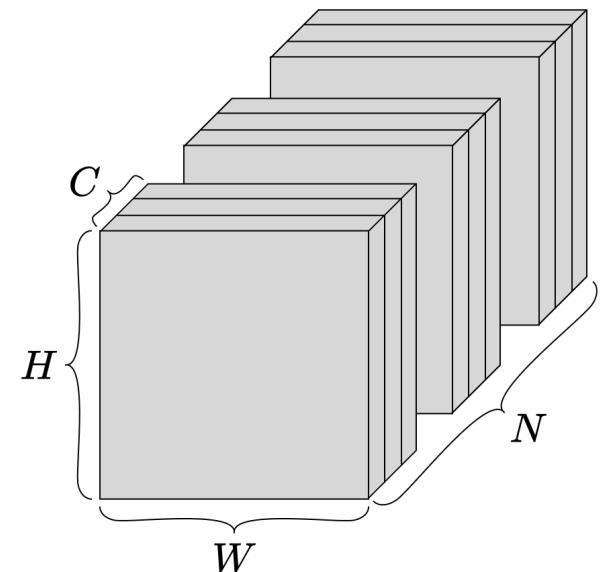
Reprezentacja danych

W widzeniu komputerowym sekwencja wideo reprezentowana jest jako tensor 4-wymiarowy:

$$I \in \mathbb{R}^{N \times C \times H \times W} \quad (3)$$

Gdzie wymiary oznaczają kolejno:

- N : Liczba klatek,
- C : Kanały kolorów (zazwyczaj 3 dla RGB),
- H, W : Wysokość i szerokość (rozdzielcość).



Wyzwanie obliczeniowe

Przetwarzanie wideo wiąże się z rzędem wielkości większym zużyciem pamięci niż w przypadku obrazów statycznych.

Przykład: Tylko **1 sekunda** wideo 4K UHD (30 FPS, float16) zajmuje w pamięci:

$$30 \cdot 3840 \cdot 2160 \cdot 3 \cdot 2 \text{ bajty} \approx 1.49 \text{ GB} \quad (4)$$

W przypadku modeli dyfuzyjnych, wysokie zapotrzebowanie na VRAM podczas inferencji wynika z konieczności alokacji pamięci dla wielowymiarowych map cech oraz macierzy atencji.

Podstawy teoretyczne: Transformery i dyfuzja

Vision Transformer...

Tradycyjne sieci splotowe ograniczają się do **lokalnego pola recepcji**. Wprowadzenie architektury Transformer zmieniło ten paradygmat:

1. **Tokenizacja:** Obraz dzielony jest na sekwencję łat (ang. *patches*), które traktowane są analogicznie do słów w przetwarzaniu języka.

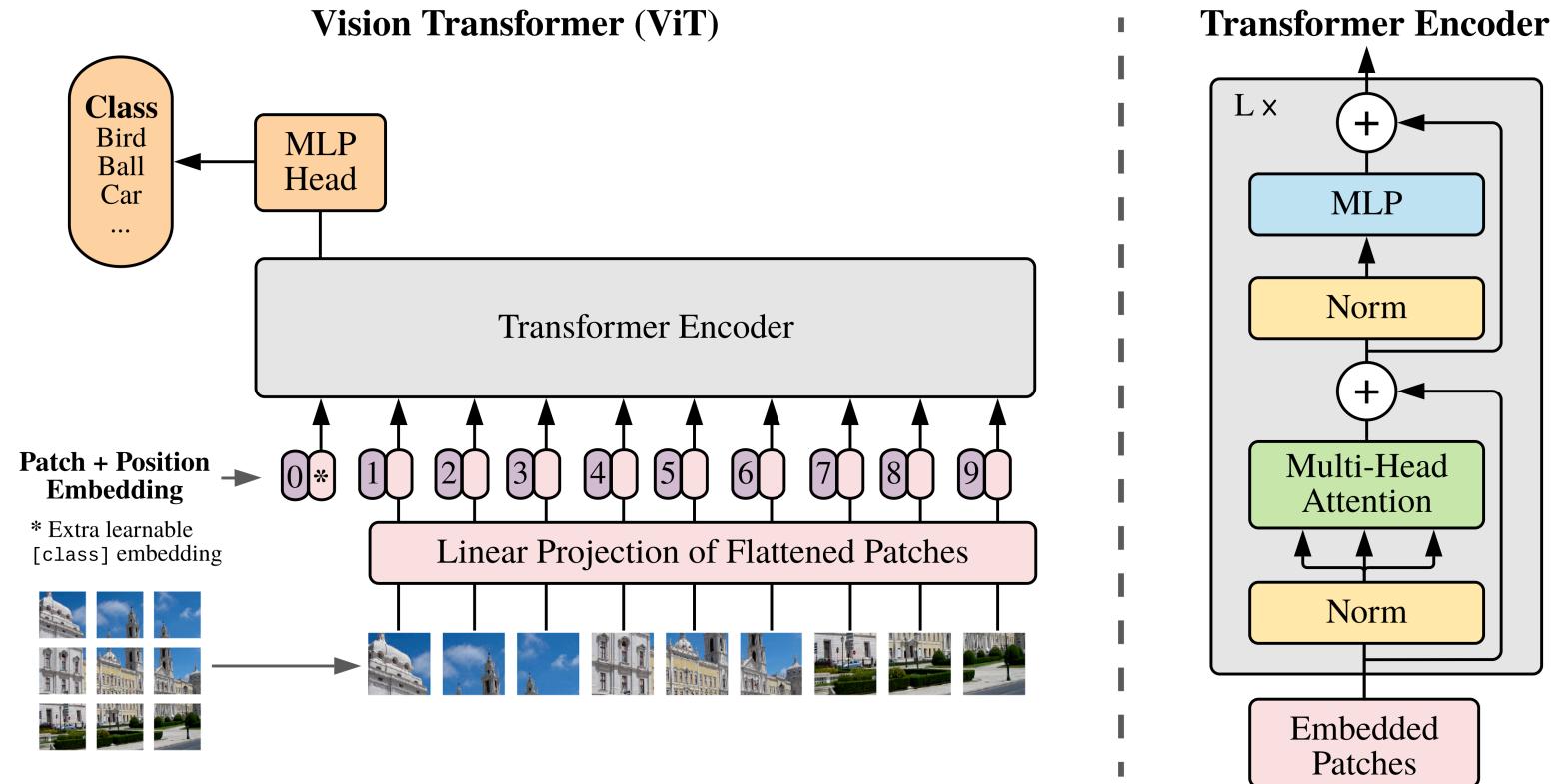
Vision Transformer...

2. **Globalny mechanizm uwagi:** Umożliwia modelowanie **długodystansowych zależności**. Każdy fragment obrazu może czerpać informacje z każdego innego, niezależnie od odległości w czasoprzestrzeni.

$$\text{Attention}(Q, K, V) = \text{softmax} \left(\frac{QK^T}{\sqrt{d_k}} \right) V \quad (5)$$

3. **Problem złożoności:** Analiza globalna wiąże się ze **złożonością kwadratową** $O(N^2)$ względem liczby tokenów. Dla wideo wysokiej rozdzielczości macierz atencji staje się wąskim gardłem pamięciowym.

Vision Transformer



Rysunek 2: Schemat architektury Vision Transformer (ViT) [2].

Generatywne modele dyfuzyjne...

Modele dyfuzyjne [3] to probabilistyczne modele generatywne, które uczą się tworzyć dane poprzez **iteracyjne odwracanie procesu degradacji**. Całość opiera się na dwóch łańcuchach Markowa:

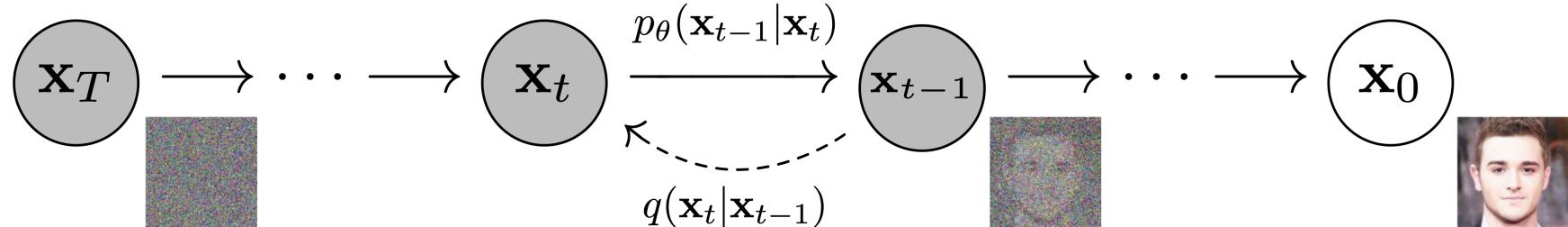
1. **Proces zaszumiania:** Polega na stopniowym, krokowym dodawaniu szumu Gaussa do obrazu wejściowego x_0 . Po wykonaniu T kroków, oryginalny obraz zamienia się w całkowity szum losowy.

$$q(x_t \mid x_{t-1}) = \mathcal{N}\left(x_t; \sqrt{1 - \beta_t}x_{t-1}, \beta_t I\right) \quad (6)$$

Generatywne modele dyfuzyjne

2. **Proces odszumiania:** To właściwy proces generacji. Sieć neuronowa uczy się przewidywać stan x_{t-1} na podstawie zaszumionego x_t .

$$p_\theta(x_{t-1} \mid x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \quad (7)$$



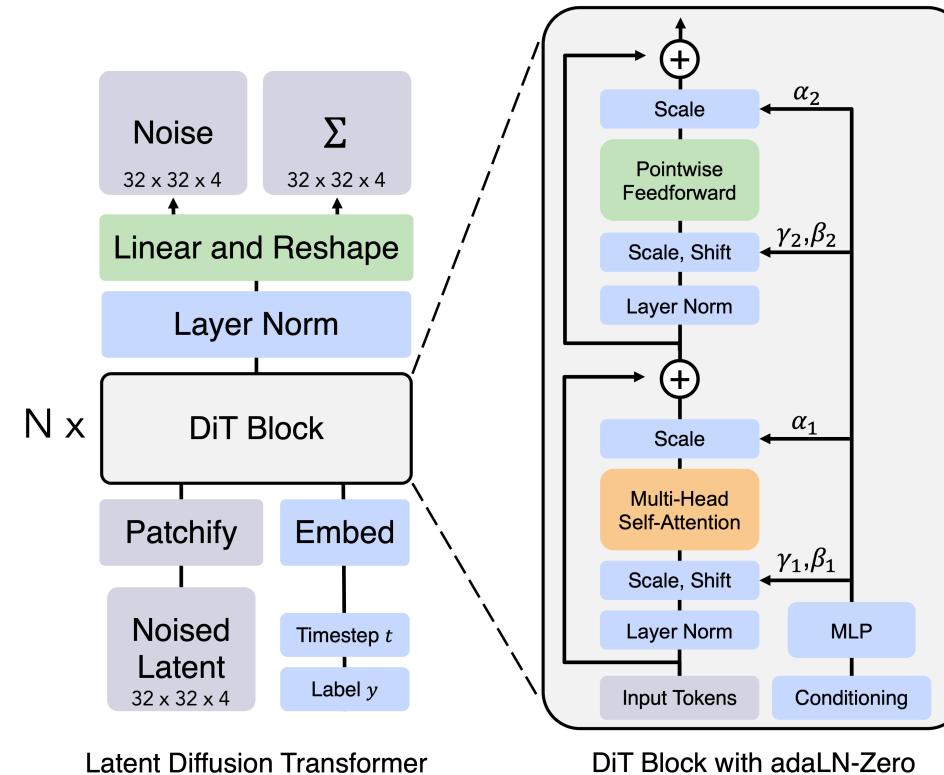
Rysunek 3: Schemat działania modelu dyfuzyjnego [3].

Architektura Diffusion Transformer...

Peebles i Xie [4] zaproponowali zastąpienie klasycznego U-Netu architekturą Transformera. Proces przetwarzania przebiega w trzech krokach:

1. **Tokenizacja:** Wejściowy zaszumiony tensor (**w przestrzeni ukrytej**) jest dzielony na sekwencję lat x_p .
2. **Bloki Transformera z mechanizmem adaLN:** Zamiast standardej normalizacji, w DiT zastosowano **Adaptive Layer Norm (adaLN)**.
3. **Projekcja końcowa i przekształcenie:** Przetworzony tensor przechodzi przez warstwę liniową, która rzutuje go do kształtu wejściowego.

Architektura Diffusion Transformer...



Rysunek 4: Schemat architektury Diffusion Transformer (DiT). Po lewej: Globalny przepływ danych. Po prawej: Szczegóły bloku DiT z mechanizmem adaLN-Zero.

Architektura Diffusion Transformer

W dotychczasowych modelach dyfuzyjnych standardem był splotowy U-Net. Zastąpienie go architekturą DiT wnosi kluczowe ulepszenia:

- **Skalowalność,**
- **Globalne przetwarzanie kontekstu,**
- **Uproszczenie architektury.**

Analiza architektury FlashVSR

FlashVSR...

FlashVSR [5] to model dyfuzyjny do VSR działający w trybie strumieniowym i generujący wynik w pojedynczym kroku.

- **Wydajność SOTA:** Model osiąga prędkość **17 FPS** dla rozdzielczości 768×1408 na pojedynczym układzie A100.
- **Generalizacja do ultra-wysokich rozdzielczości:** Dzięki unikalnej konstrukcji atencji, FlashVSR eliminuje błędy generalizacji przy skalowaniu do ultra-wysokich rozdzielczości.

FlashVSR

Fundamentem FlashVSR jest trójetapowy proces destylacji wiedzy oraz architektura przystosowana do przetwarzania przyczynowego (causal).

- **Nauczyciel i Uczeń:** Wiedza z potężnego modelu nauczyciela jest destylowana do lekkiego modelu ucznia.
- **Przetwarzanie strumieniowe (KV Cache):** Model wykorzystuje mechanizm **KV Cache**, znany z dużych modeli językowych.

Innowacje architektury FlashVSR

1. Locality-Constrained Sparse Attention

- **Ograniczenie lokalne:** Wymuszenie atencji w lokalnym oknie eliminuje błędy pozycyjne i zapobiega „zawijaniu się” wzorców,
- **Rzadka atencja:** Przetwarzanie jest wyłącznie **top-k** kluczowych obszarów.

2. Tiny Conditional Decoder

- **Warunkowanie klatką LR:** Bezpośrednie wykorzystanie klatki niskiej rozdzielczości jako sygnału pomocniczego upraszcza zadanie sieci.

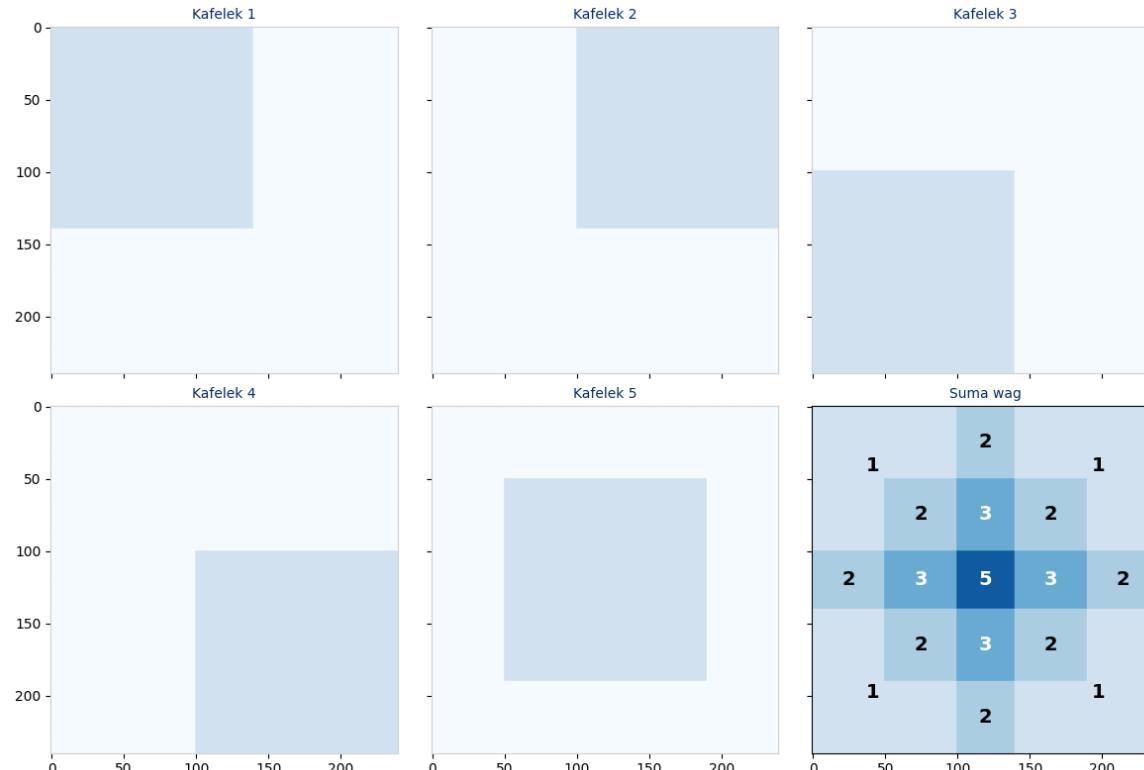
Optymalizacja i implementacja

Implementacja potoku przetwarzania...

W celu uruchomienia modelu na kartach graficznych klasy konsumenckiej, zaimplementowałem potok przetwarzania wykorzystujący techniki kafelkowania:

1. **Kafelkowanie czasowe:** Wideo jest przetwarzana sekwencyjnie w krótszych klipach.
2. **Kafelkowanie przestrzenne:** Każda klatka dzielona jest na mniejsze fragmenty z uwzględnieniem marginesu.

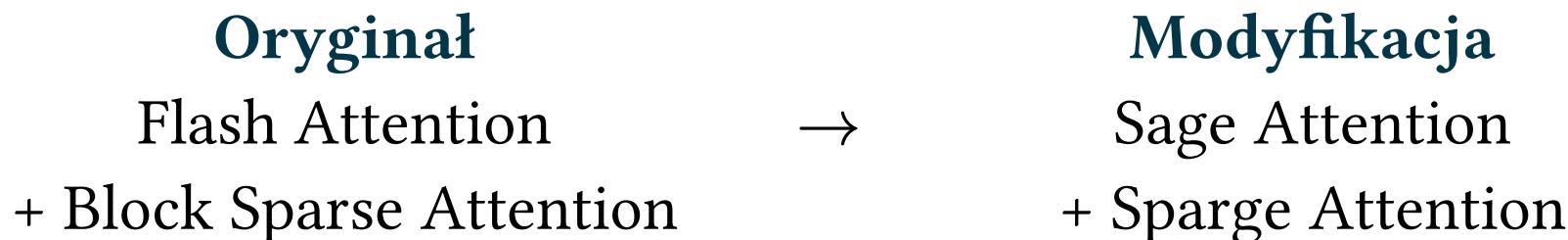
Implementacja potoku przetwarzania



Rysunek 5: Przykład kafelkowania przestrzennego.

Optymalizacja mechanizmu atencji

Istotną modyfikacją względem oryginalnej architektury była wymiana kerneli obliczeniowych atencji:



1. **Sage Attention:** Zastosowanie precyzyjnej kwantyzacji 8-bitowej (int8) w macierzach atencji.
2. **Spurge Attention:** Zoptymalizowany wariant rzadkiej atencji, redukujący złożoność obliczeniową dla tokenów o niskiej istotności.

Evaluacja i podsumowanie

Wyniki eksperymentów

Tabela 1: Porównanie jakości rekonstrukcji dla trzech badanych konfiguracji. Dla metod wykorzystujących kafelkowanie przestrzenne przyjęto parametry: rozmiar kafelka 192×192 oraz margines 24 px.

Zbiór danych	Metryka	FlashVSR	FlashVSR + kafelkowanie	FlashVSR + kafelkowanie + modyfikacja atencji
REDS	PSNR↑	23.31	23.16	23.13
	SSIM↑	0.6110	0.6075	0.6068
	LPIPS↓	0.3866	0.3950	0.3962
	NIQE↓	3.489	3.580	3.595
	MUSIQ↑	66.63	65.20	65.05
	CLIPQA↑	0.5221	0.5160	0.5152
	DOVER↑	12.66	12.15	12.08
VideoLQ	NIQE↓	4.070	4.150	4.165
	MUSIQ↑	52.27	51.40	51.28
	CLIPQA↑	0.3601	0.3540	0.3532
	DOVER↑	7.481	7.250	7.210

Prezentacja efektów wizualnych

Poniżej przedstawiono bezpośrednie porównanie sekwencji wejściowej z wynikiem rekonstrukcji uzyskanym przez model FlashVSR.



Rysunek 6: Wideo LR.



Rysunek 7: Wideo HR (FlashVSR).

Plan prac

W ramach pracy przewidziałem realizację następujących etapów:

1. Analiza technik kwantyzacji:

- **PTQ (Post-Training Quantization):** Kwantyzacja wytrenowanego modelu.
- **QAT (Quantization-Aware Training):** Integracja kwantyzacji podczas treningu modelu.

2. Implementacja aplikacji użytkowej:

Stworzenie aplikacji z graficznym interfejsem, który zintegruje opracowany potok przetwarzania.

Bibliografia

- [1] H. Liu *et al.*, „Video super-resolution based on deep learning: a comprehensive survey”, *Artificial Intelligence Review*, t. 55, nr 8, s. 5981–6035, 2022.
- [2] A. Dosovitskiy *et al.*, „An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale”. [Online]. Dostępne na: <https://arxiv.org/abs/2010.11929>
- [3] J. Ho, A. Jain, i P. Abbeel, „Denoising Diffusion Probabilistic Models”. [Online]. Dostępne na: <https://arxiv.org/abs/2006.11239>
- [4] W. Peebles i S. Xie, „Scalable Diffusion Models with Transformers”. [Online]. Dostępne na: <https://arxiv.org/abs/2212.09748>
- [5] J. Zhuang *et al.*, „FlashVSR: Towards Real-Time Diffusion-Based Streaming Video Super-Resolution”. [Online]. Dostępne na: <https://arxiv.org/abs/2510.12747>
- [6] A. A. Baniya, T.-K. Lee, P. W. Eklund, i S. Aryal, „A Survey of Deep Learning Video Super-Resolution”, *IEEE Transactions on Emerging Topics in Computational Intelligence*, t. 8, nr 4, s. 2655–2676, 2024, doi: 10.1109/TETCI.2024.3398015.