Wielowarstwowe perceptrony operujące na wycinkach obrazów

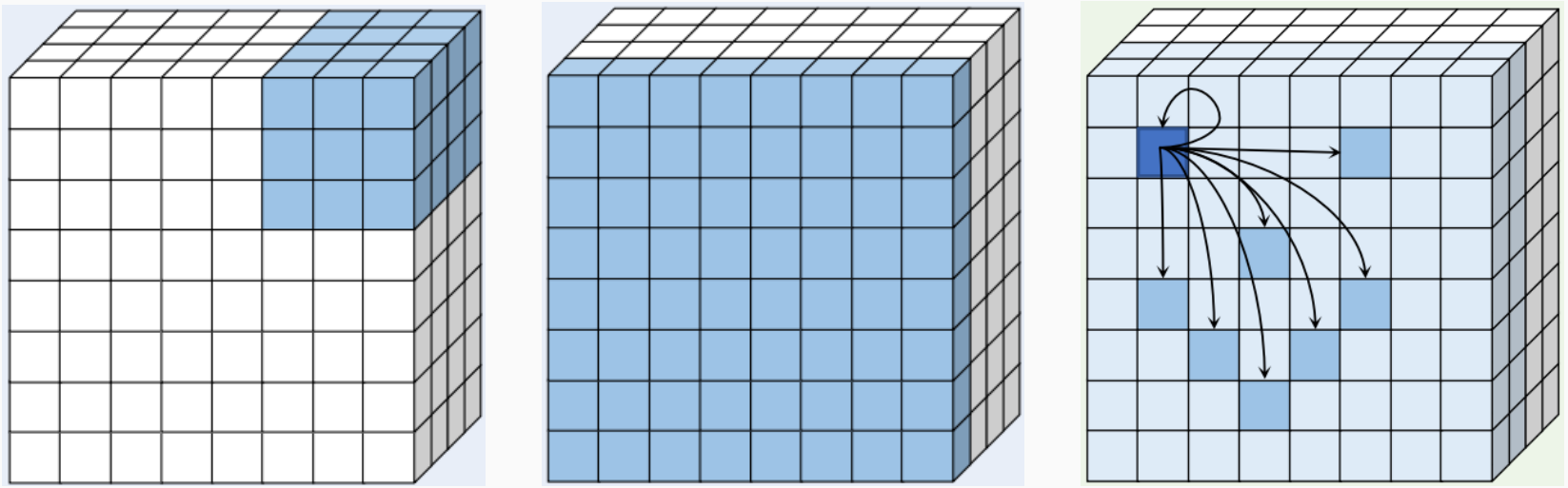
Mikołaj Małkiński
m.malkinski@mini.pw.edu.pl

29 czerwca 2022

Politechnika Warszawska
Szkoła Doktorska nr 3

Modele operują na macierzach $X \in \mathbb{R}^{H \times W \times C}$, gdzie:

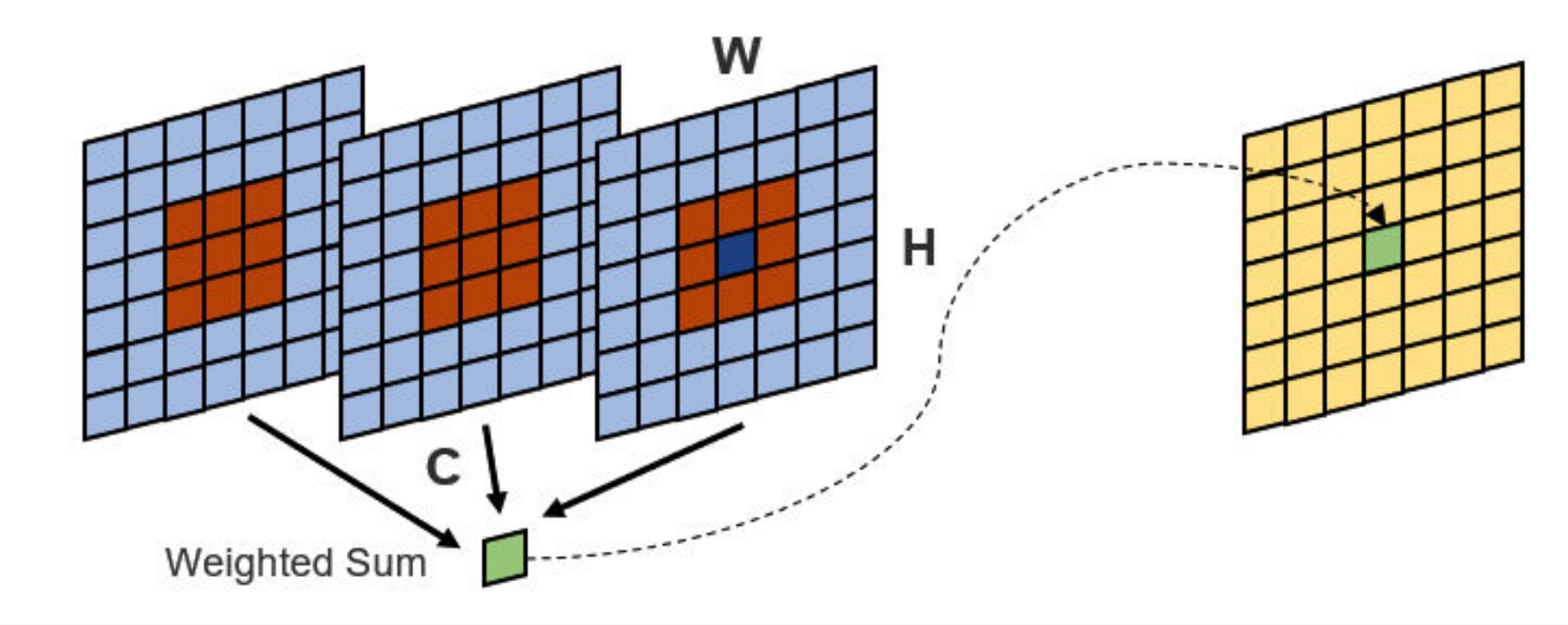
- H - wysokość (height)
- W - szerokość (width)
- C - głębokość (number of channels)



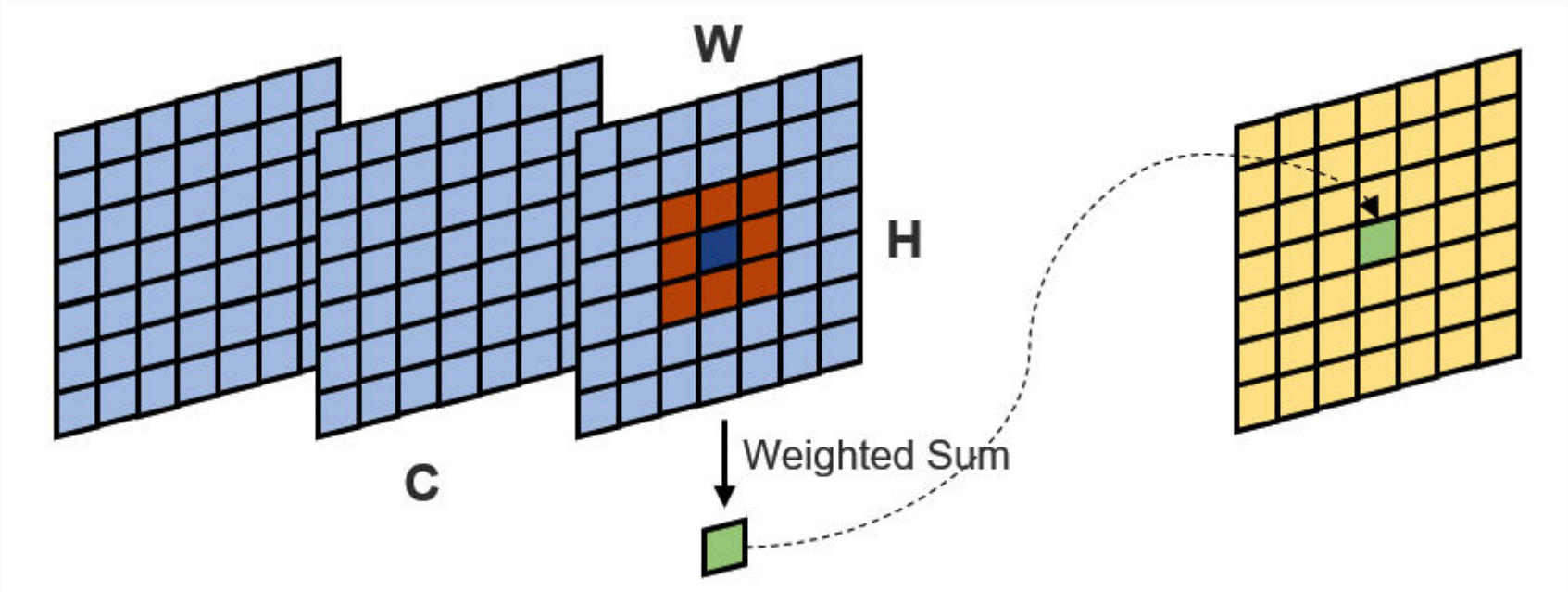
(a) Convolution **(b)** Dense **(c)** Self-attention.

Rysunek 1: Rysunki z [15].

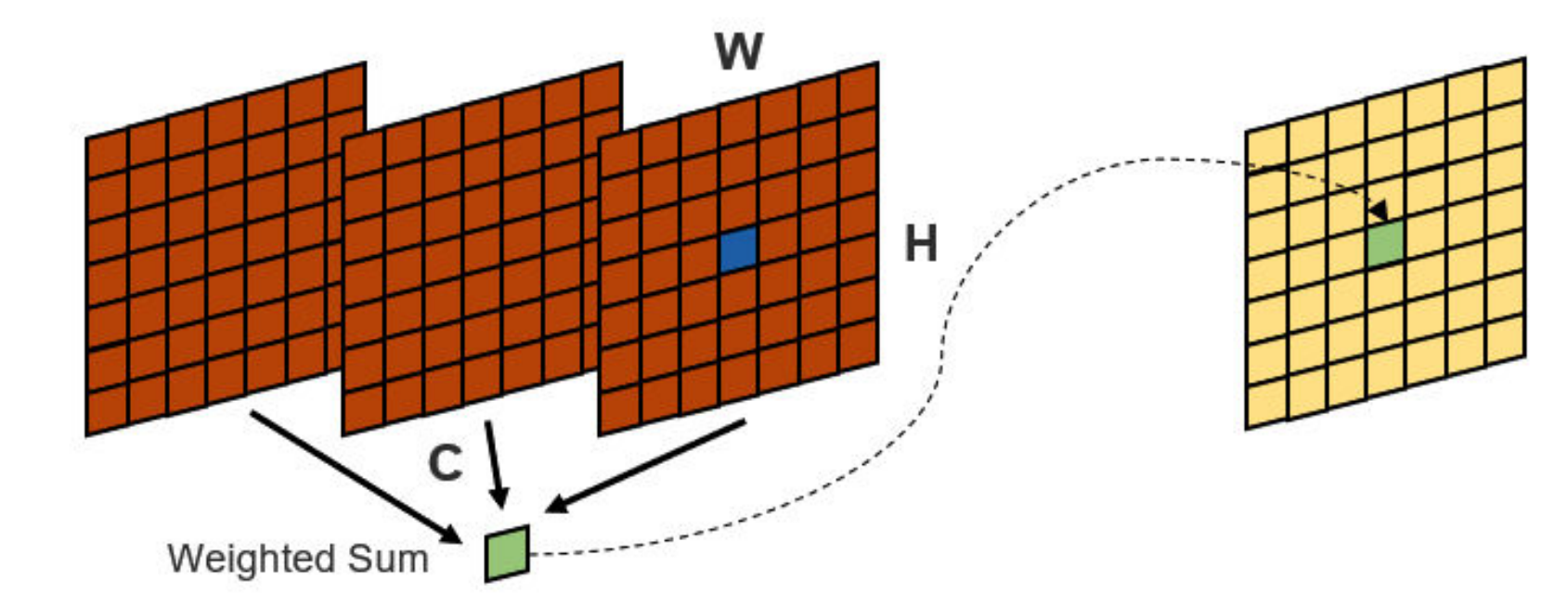
Liu, Ruiyang, Yinghui Li, Linmi Tao, Dun Liang, Shi-Min Hu, and Hai-Tao Zheng. **"Are we ready for a new paradigm shift? A Survey on Visual Deep MLP."** arXiv preprint arXiv:2111.04060 (2021).



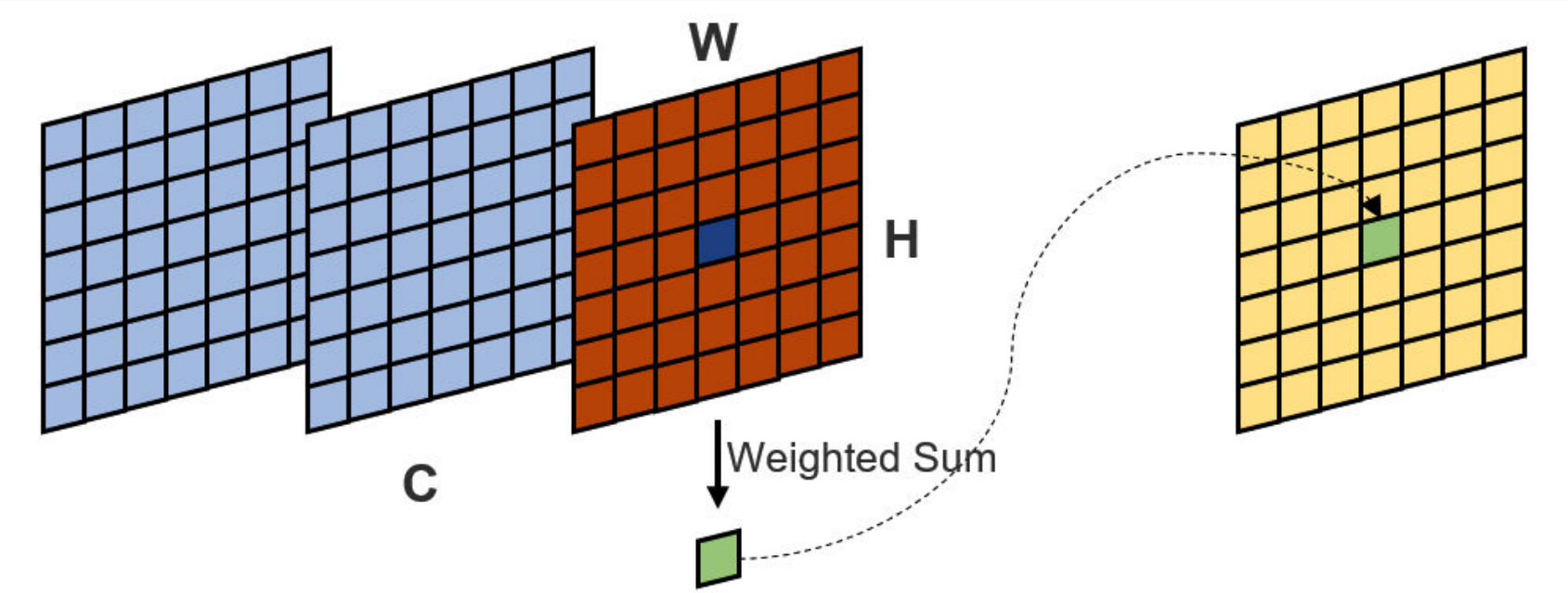
Rysunek 2: Convolutional filter. Rysunek z [7].



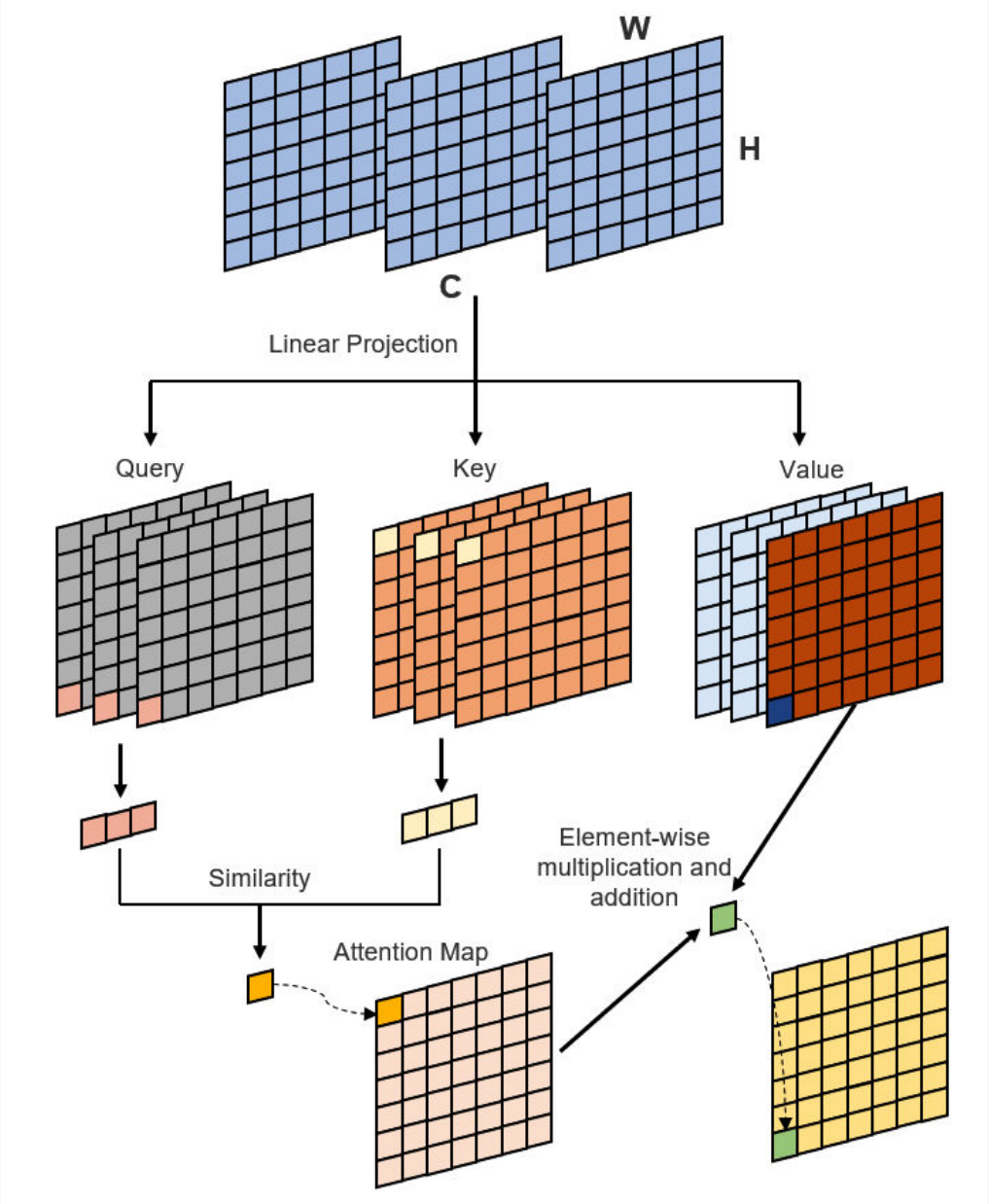
Rysunek 3: Depth-wise convolutional filter. Rysunek z [7].



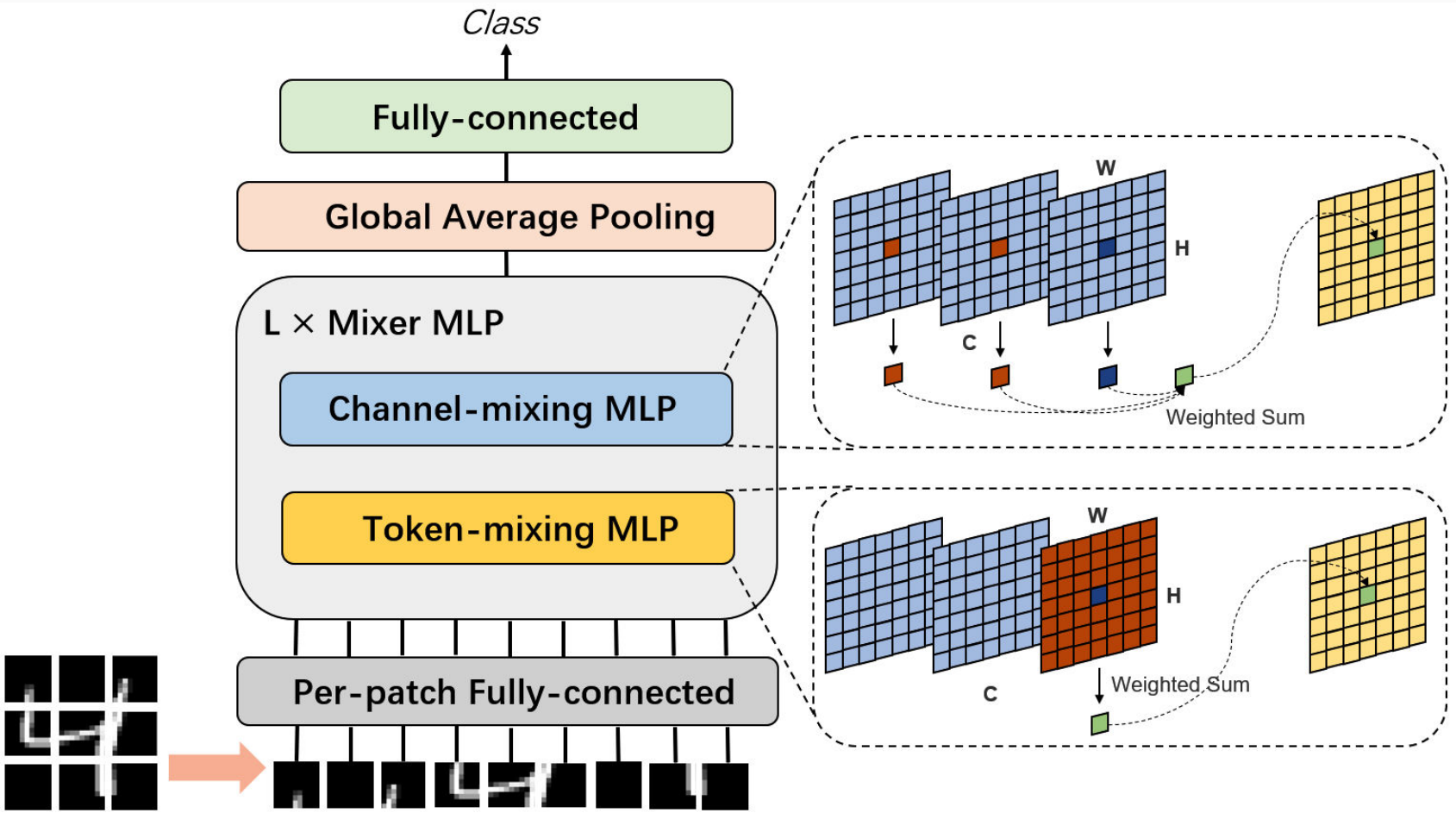
Rysunek 4: Multi-layer perceptron (MLP). Rysunek z [7].



Rysunek 5: Token mixer MLP. Rysunek z [7].



Rysunek 6: Self-attention. Rysunek z [7].



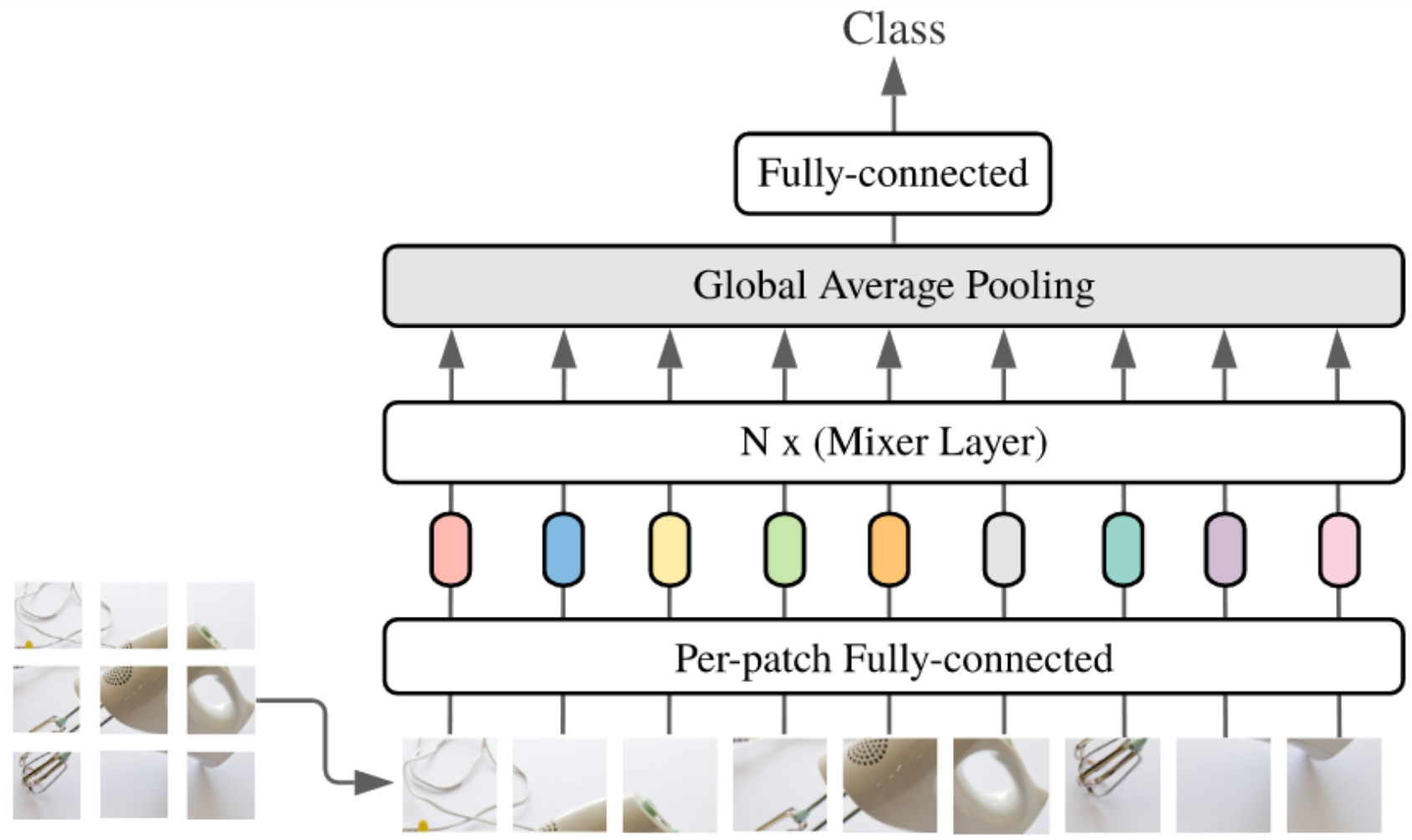
Rysunek 7: MLP-mixer [11]. Rysunek z [7].

Rozważmy funkcję $f : X \to y$, gdzie:

- $X \in [0, 1]^{H \times W \times 3}$ – kolorowy obrazek o wysokości H i szerokości W
- $y \in \mathbb{Z}$ – etykieta (np. indeks klasy)

Rozbijmy obrazek na wycinki/fragmenty (ang. patch)
$X = \{\{\mathrm{p}_{i,j}\}_{i=1}^{N_R}\}_{j=1}^{N_C}$, gdzie:

- $\mathrm{p}_{i,j} = (X_{kl})_{i*h \leq k \leq (i+1)*h \ \wedge \ j*w \leq l \leq (j+1)*w}$
- $\mathrm{p}_{i,j} \in [0,1]^{h \times w \times 3}$
- h – wysokość wycinka
- w – szerokość wycinka
- $N_R = H/h$ – liczba rzędów
- $N_C = W/w$ – liczba kolumn
- $n = N_R \cdot N_C$ – łączna liczba wycinków



Rysunek 8: MLP-mixer. Rysunek z [11].

Zbudujmy model operujący na wycinkach, o ogólnej formie:

$$f(\{\{\mathsf{p}_{i,j}\}_{i=1}^{N_R}\}_{j=1}^{N_C}) = \widehat{\mathsf{y}} \tag{1}$$

Model f będzie złożony z 3 komponentów.

Enkoder $\mathcal{E}$ tworzy reprezentację wycinków będących wektorami d elementowymi:

$$\mathsf{v}_{i,j} = \mathcal{E}(\mathrm{flat}(\mathsf{p}_{i,j})) \in \mathbb{R}^d \tag{2}$$

Rdzeń $\mathcal{C}$ złożony z m warstw iteracyjnie przetwarza dane:

$$\mathsf{V}' = \mathcal{C}_m(\ldots(\mathcal{C}_2(\mathcal{C}_1(\mathsf{V})))) \in \mathbb{R}^{n \times d} \tag{3}$$

Dekoder $\mathcal{D}$ generuje wektor logitów z liczbą elementów równą liczbie klas c:

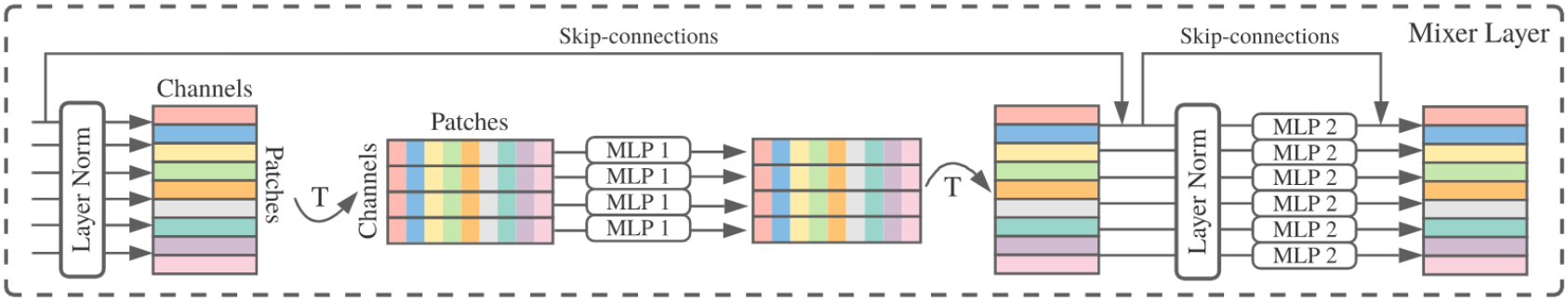
$$\mathsf{V}'' = \mathcal{D}(\mathrm{Pool}_{\mathrm{AVG}}(\mathsf{V}')) \in \mathbb{R}^{n \times c} \tag{4}$$

Otrzymaną reprezentację zamieniamy na rozkład prawdopodobieństwa klas $\widehat{\mathsf{p}}$:

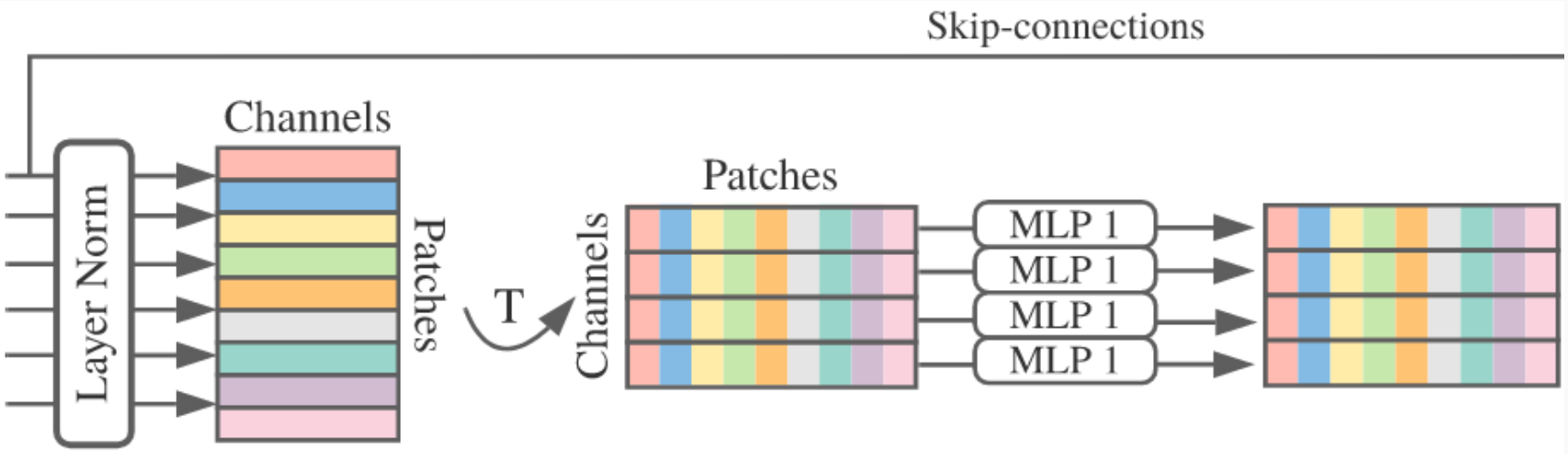
$$\widehat{\mathsf{p}} = \mathrm{softmax}(V'') \in [0,1]^c \tag{5}$$

Klasa z największym prawdopodobieństwem stanowi odpowiedź modelu:

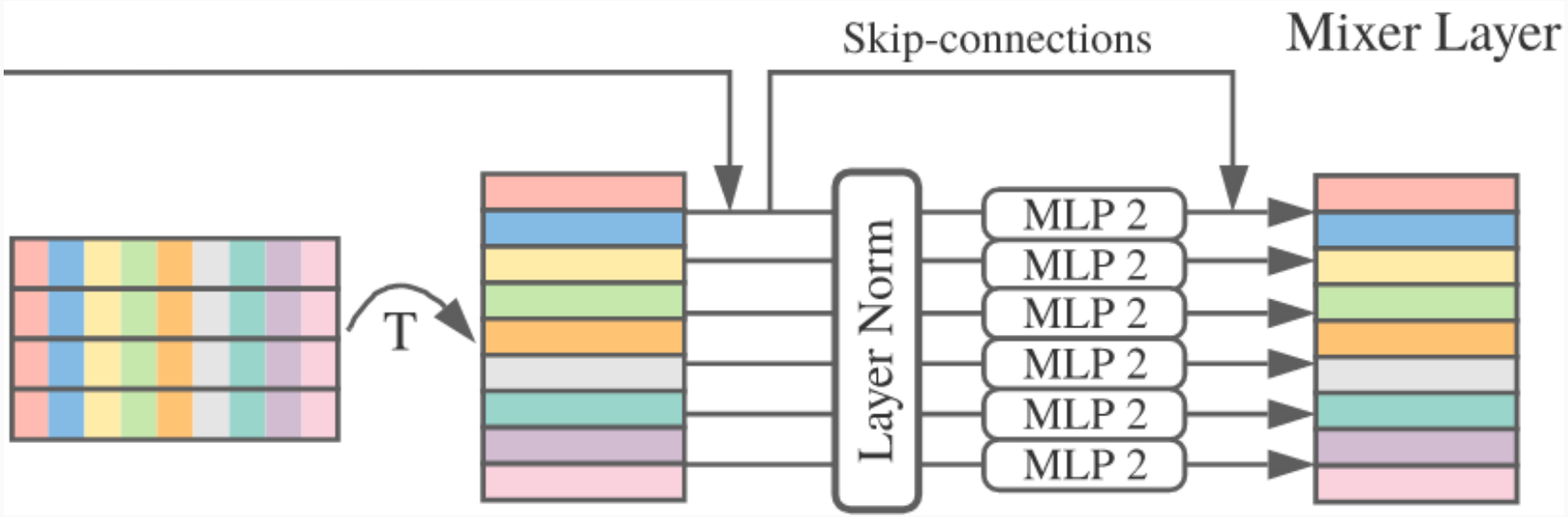
$$\widehat{\mathsf{y}} = \underset{i}{\mathsf{argmax}} \{\widehat{\mathsf{p}}_i\}_{i=1}^{c} \tag{6}$$



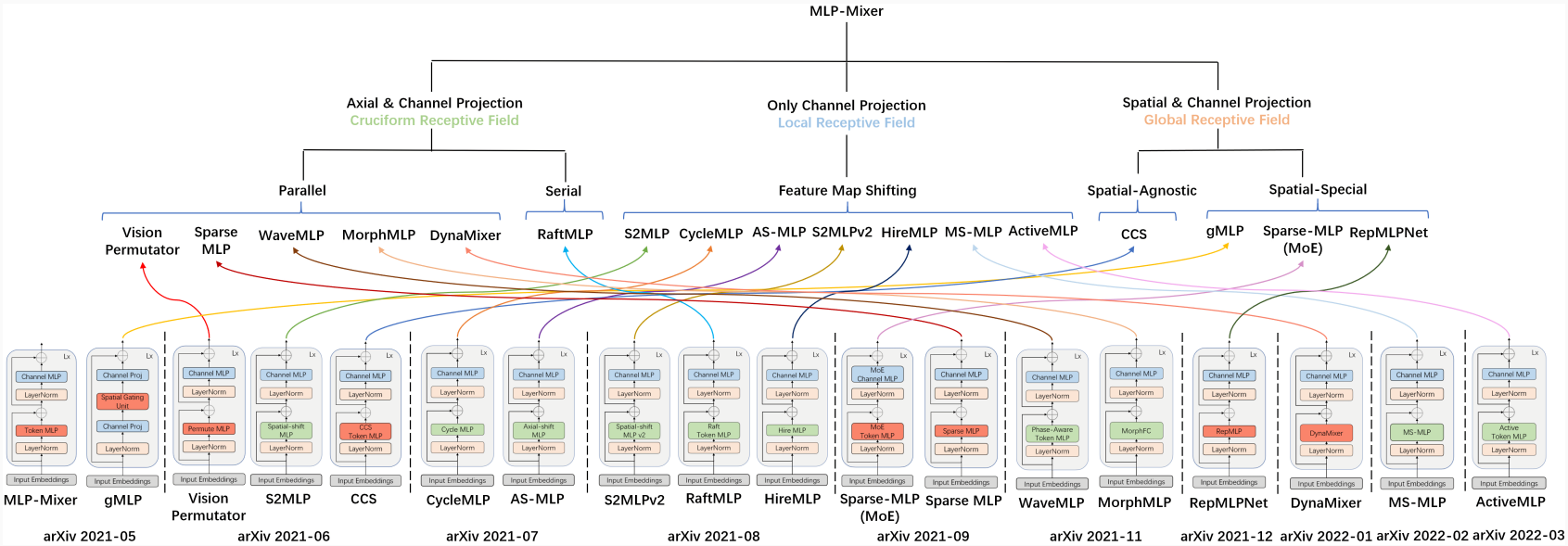
Rysunek 9: Rdzeń modelu MLP-mixer. Rysunek z [11].



Rysunek 10: Rdzeń modelu MLP-mixer (token mixing). Rysunek z [11].



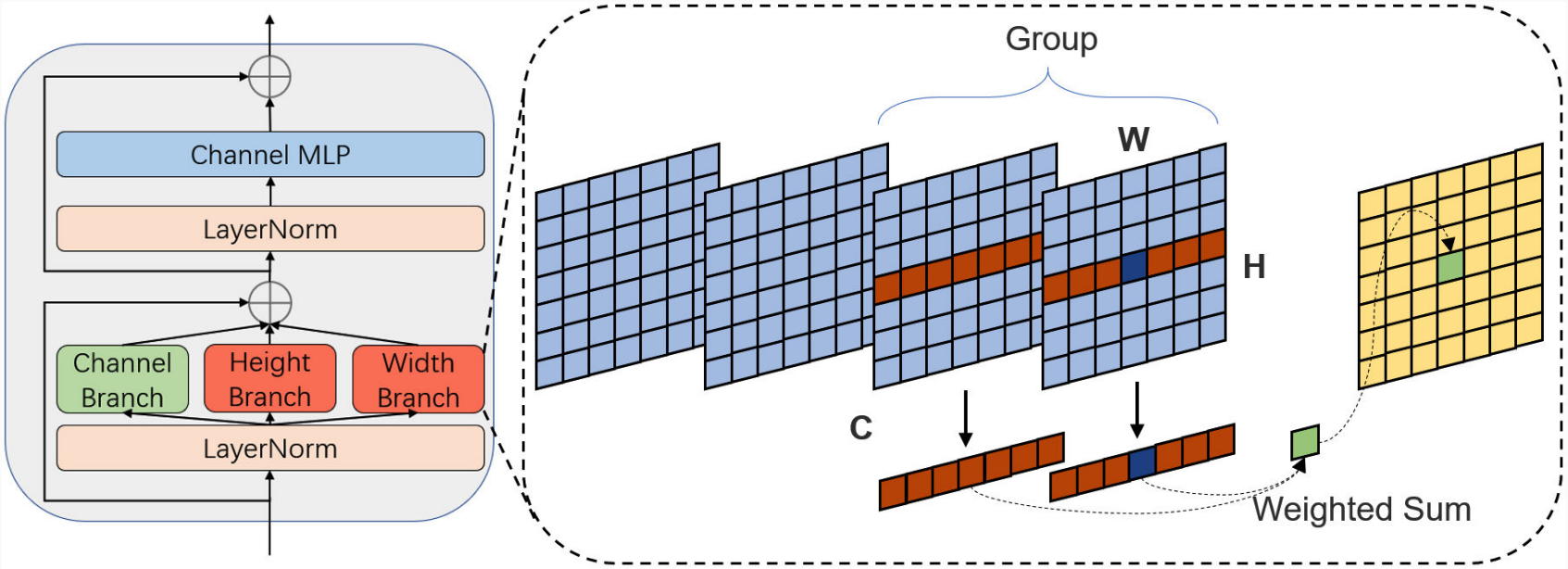
Rysunek 11: Rdzeń modelu MLP-mixer (channel mixing). Rysunek z [11].



Rysunek 12: Warianty modelu MLP-Mixer. Rysunek z [7].

Główne punkty przy konstrukcji modeli operujących na wycinkach obrazków:

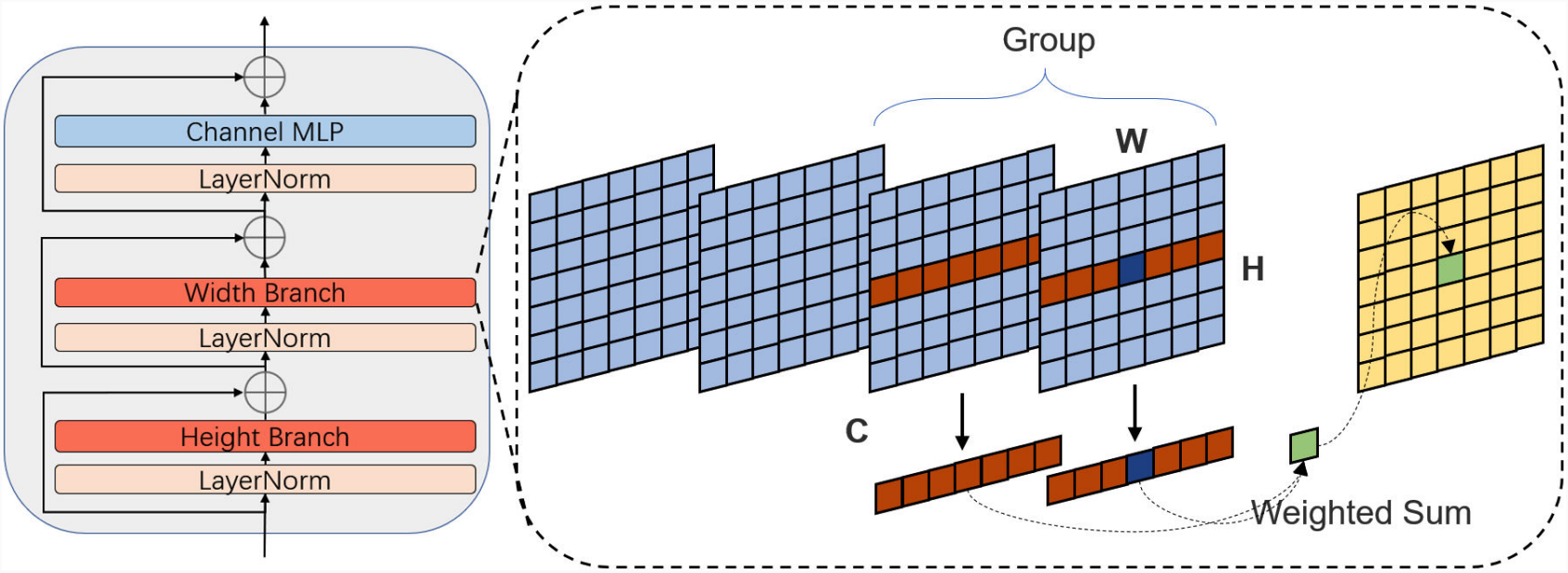
- Uwzględnienie lokalnego i globalnego kontekstu
- Obsługa obrazków o różnych wymiarach
- Receptive field



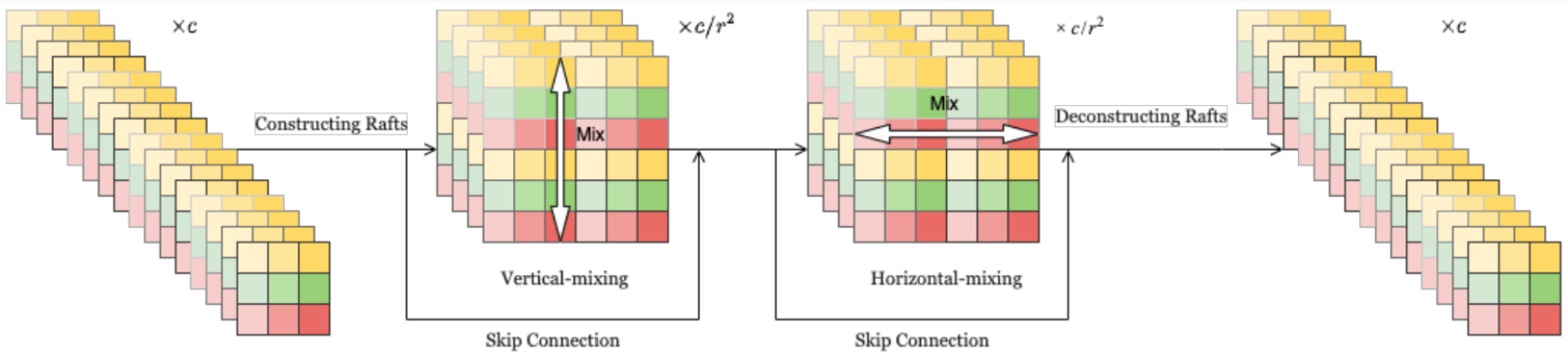
Rysunek 13: Vision Permutator (ViP) [4]. Rysunek z [7].

Operacja permutacji:

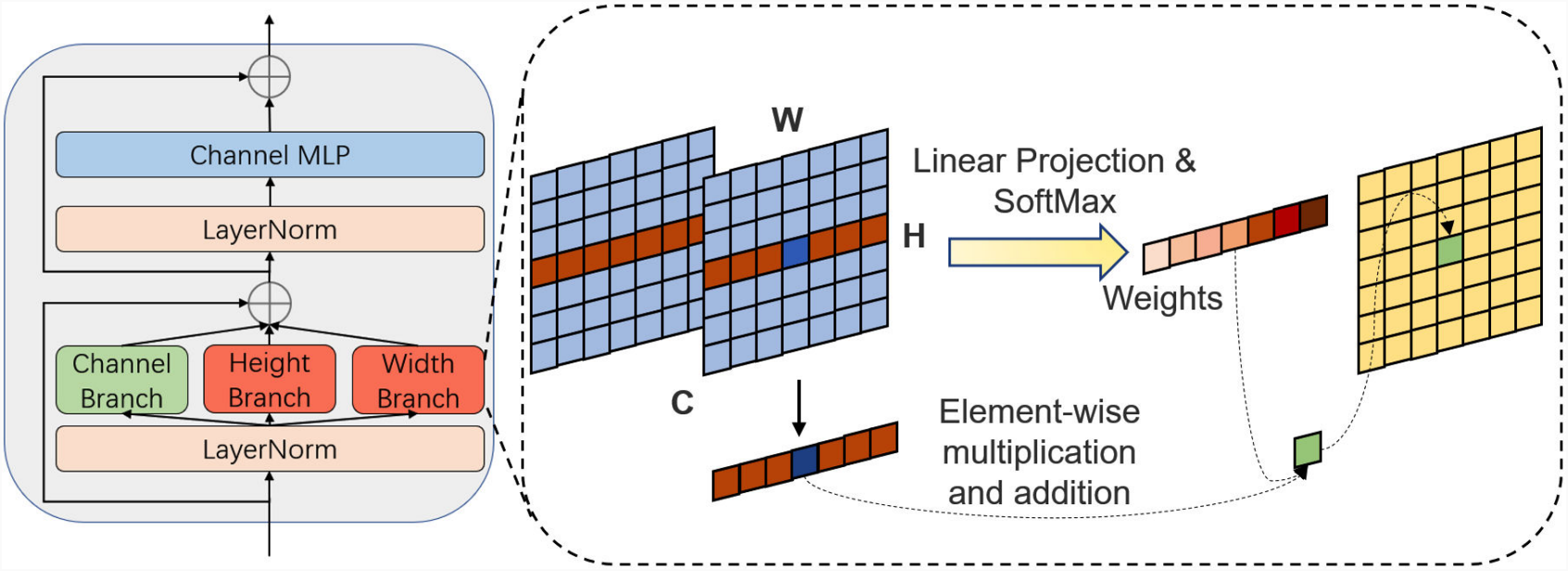
1. $H \times W \times C \rightarrow H \times W \times N \times S$
2. $H \times W \times N \times S \rightarrow S \times H \times W \times N$
3. $S \times H \times W \times N \rightarrow S \times H \times W \cdot N$



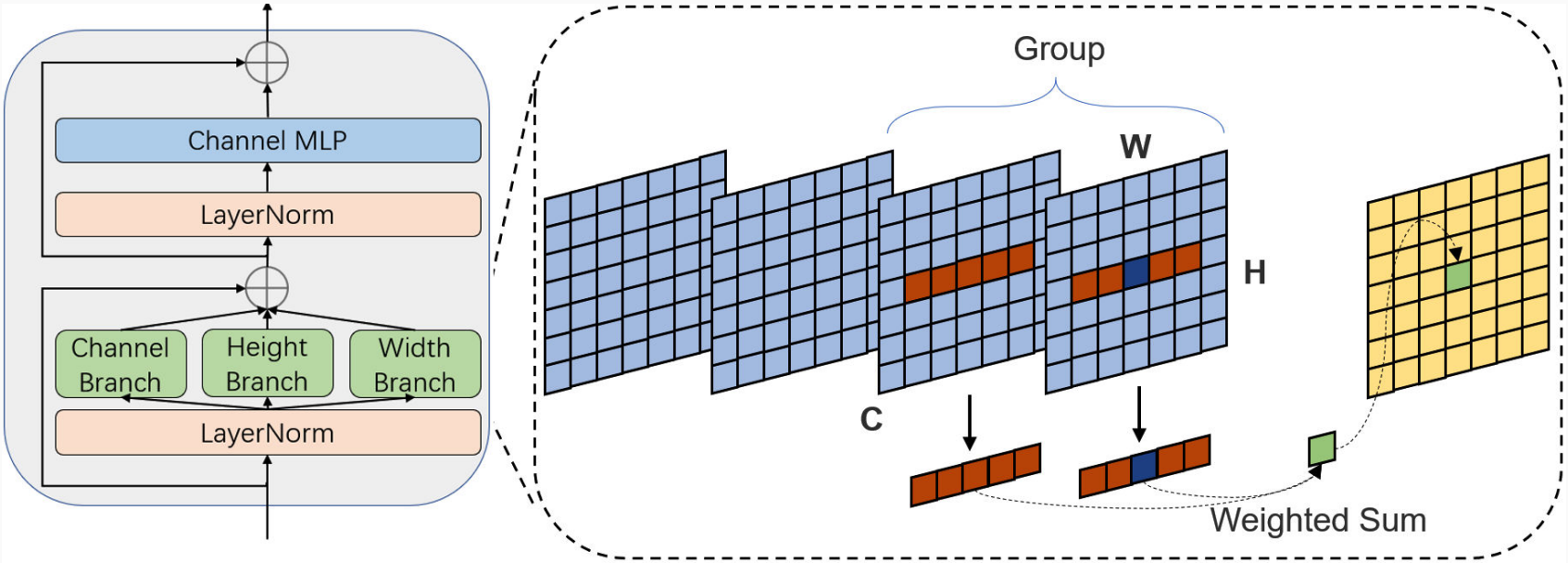
Rysunek 14: RaftMLP [10]. Rysunek z [7].



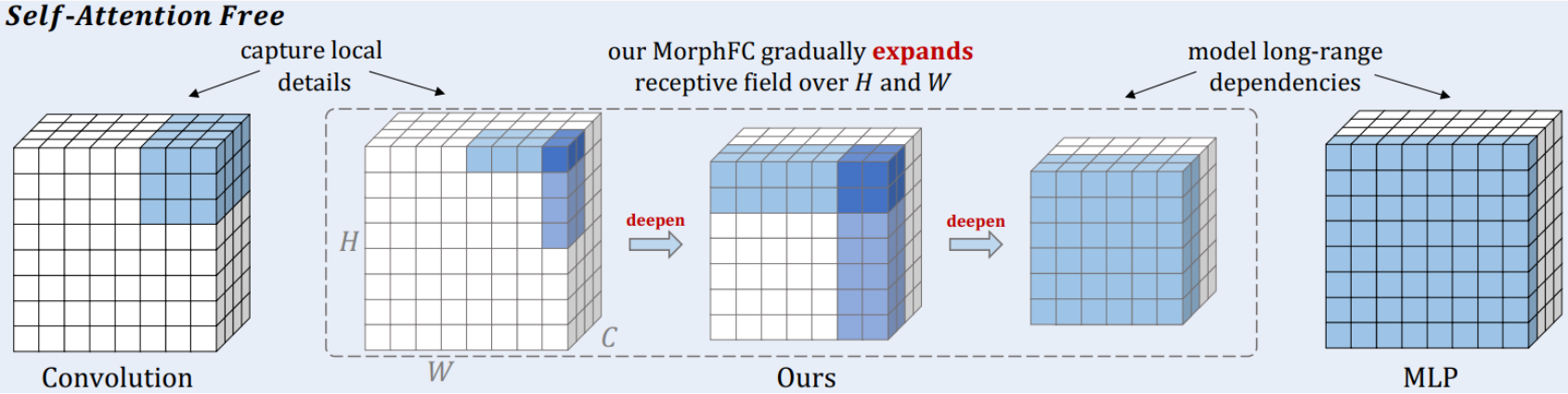
Rysunek 15: RaftMLP. Rysunek z [10].



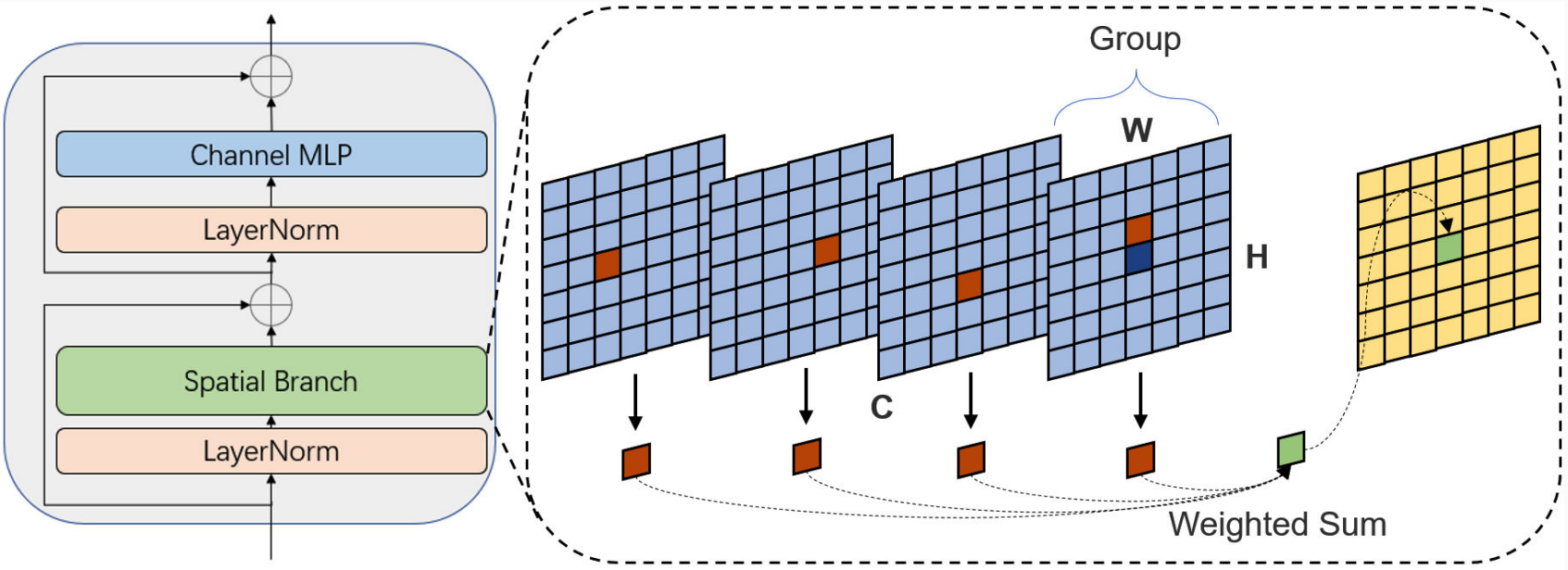
Rysunek 16: DynaMixer [12]. Rysunek z [7].



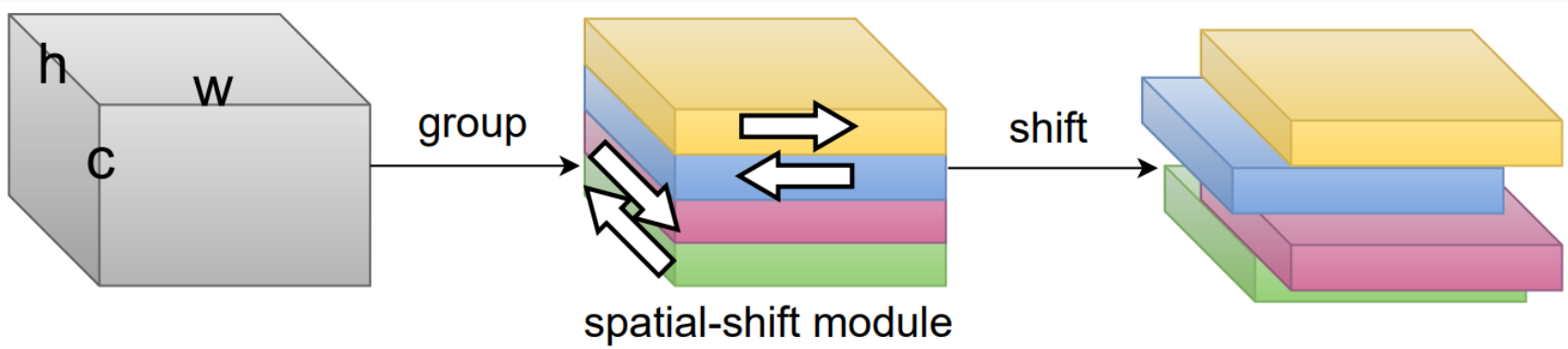
Rysunek 17: MorphMLP [15]. Rysunek z [7].



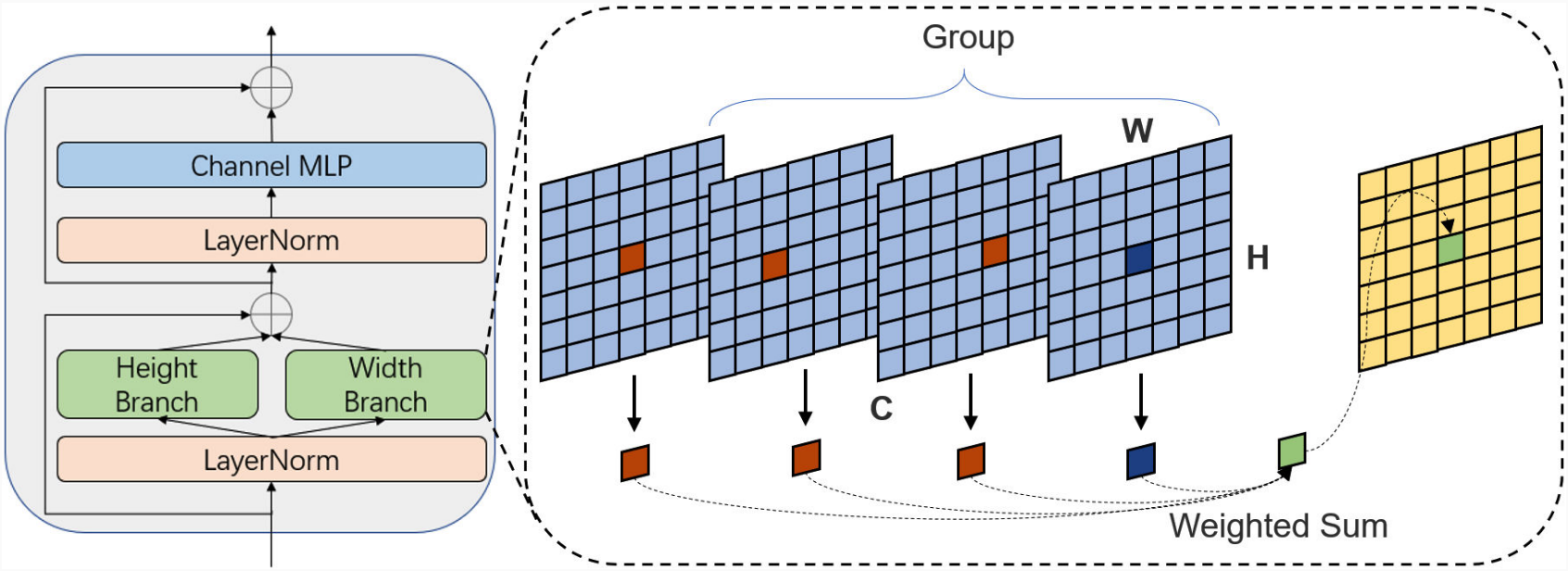
Rysunek 18: MorphMLP. Rysunek z [15].



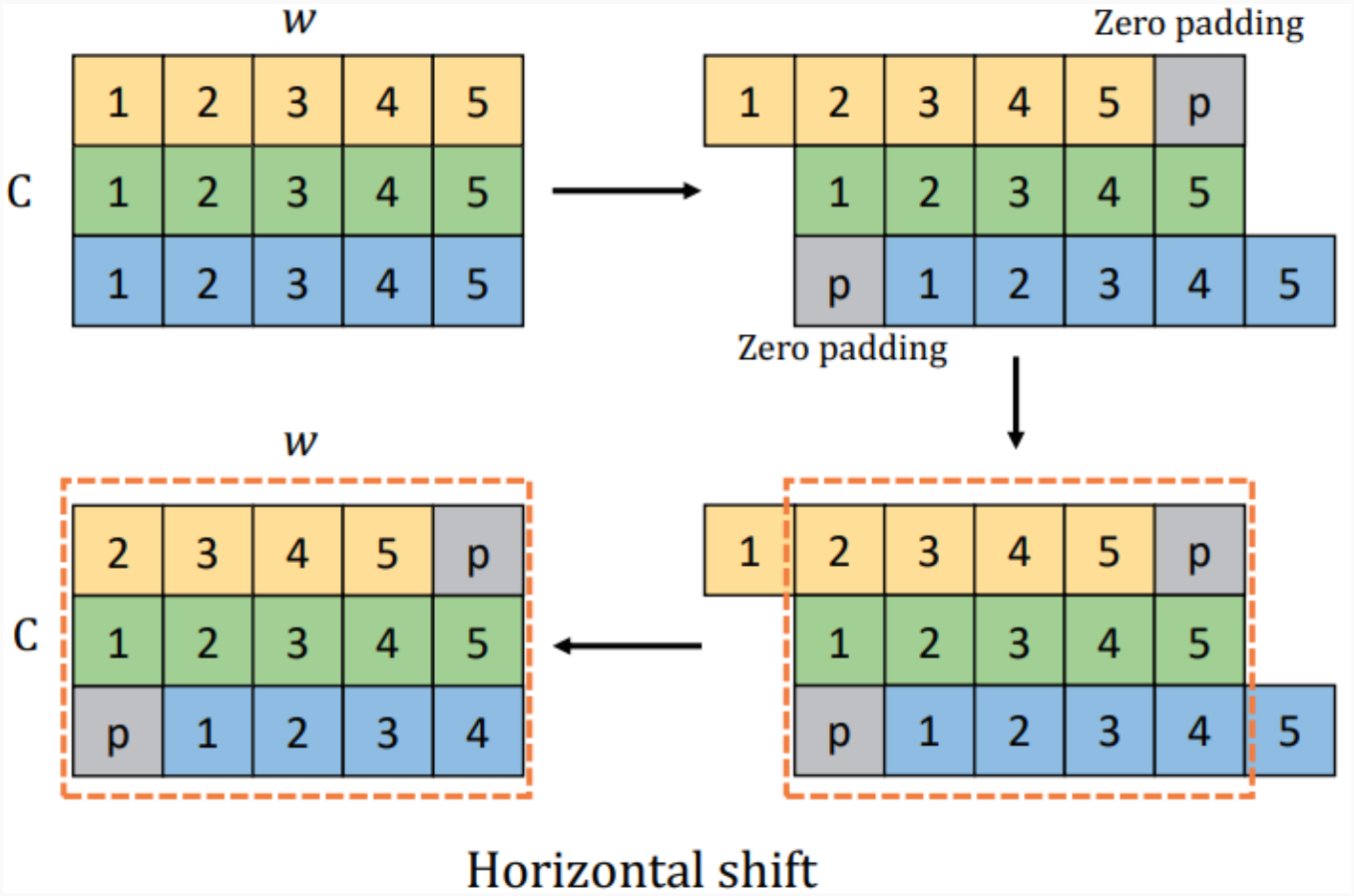
Rysunek 19: Spatial-Shift MLP (S^2-MLP) [14]. Rysunek z [7].



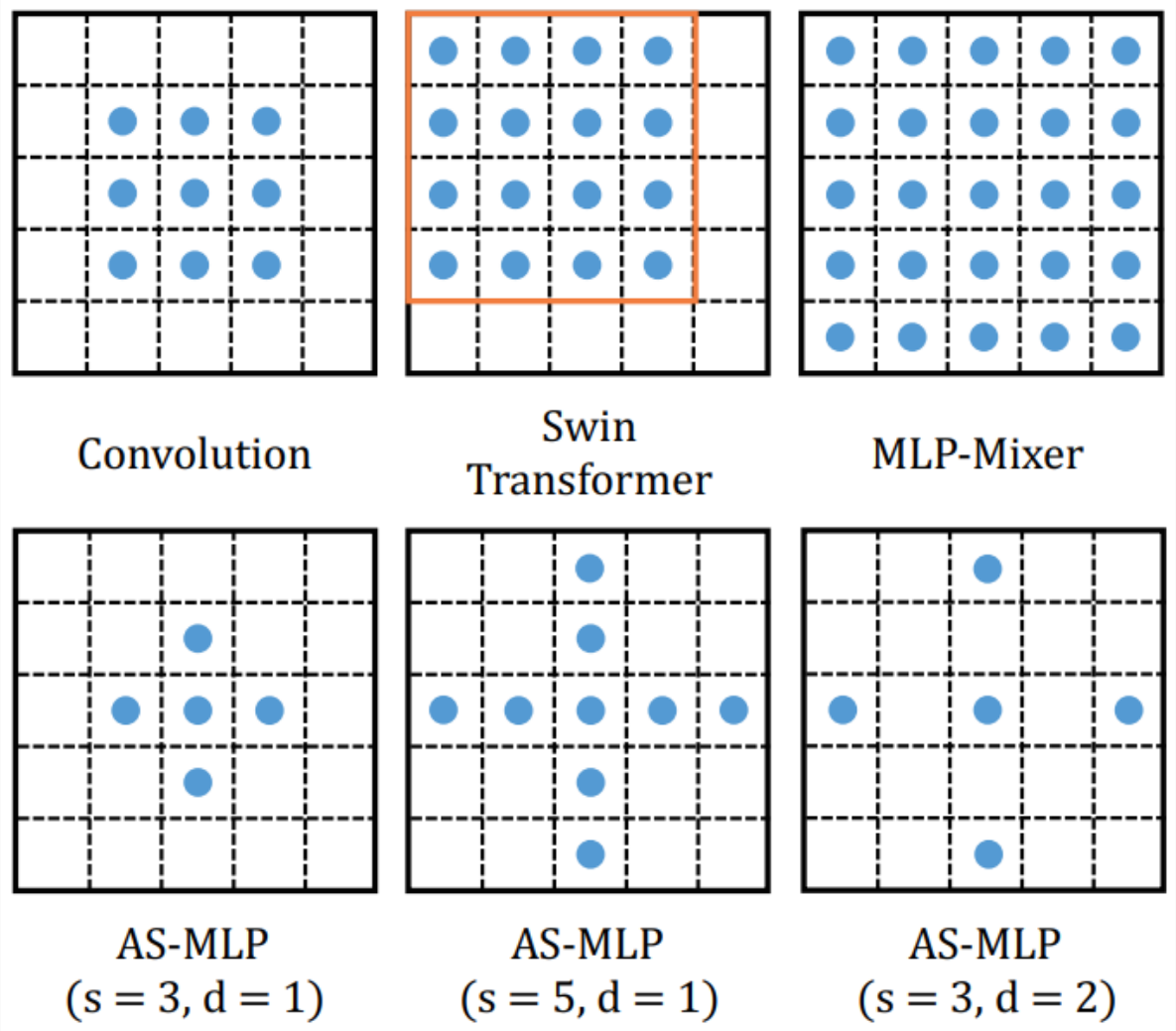
Rysunek 20: S^2-MLP. Rysunek z [14].



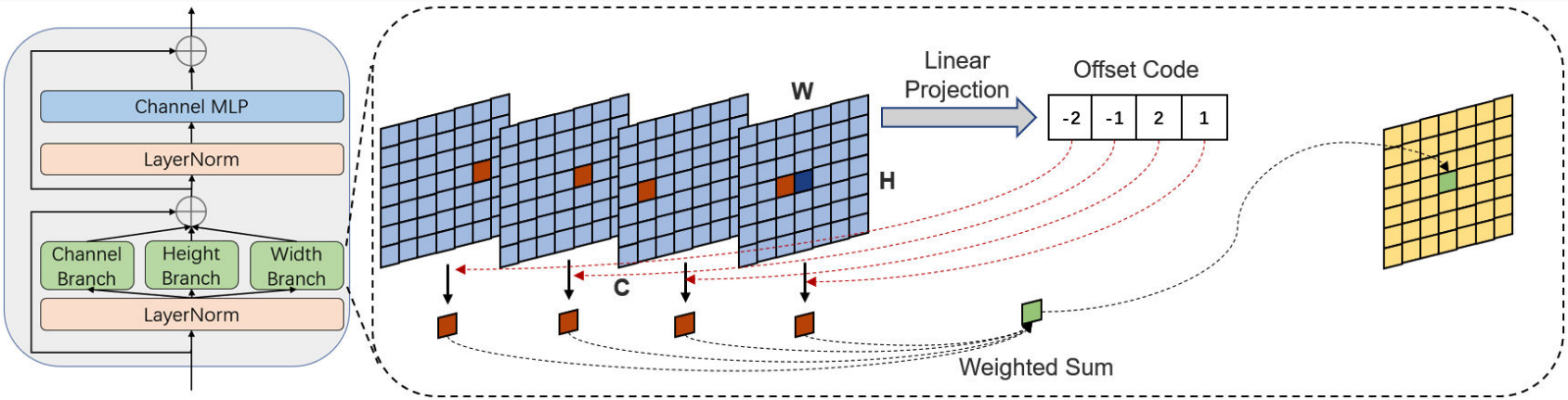
Rysunek 21: Axial-Shift MLP (AS-MLP) [6]. Rysunek z [7].



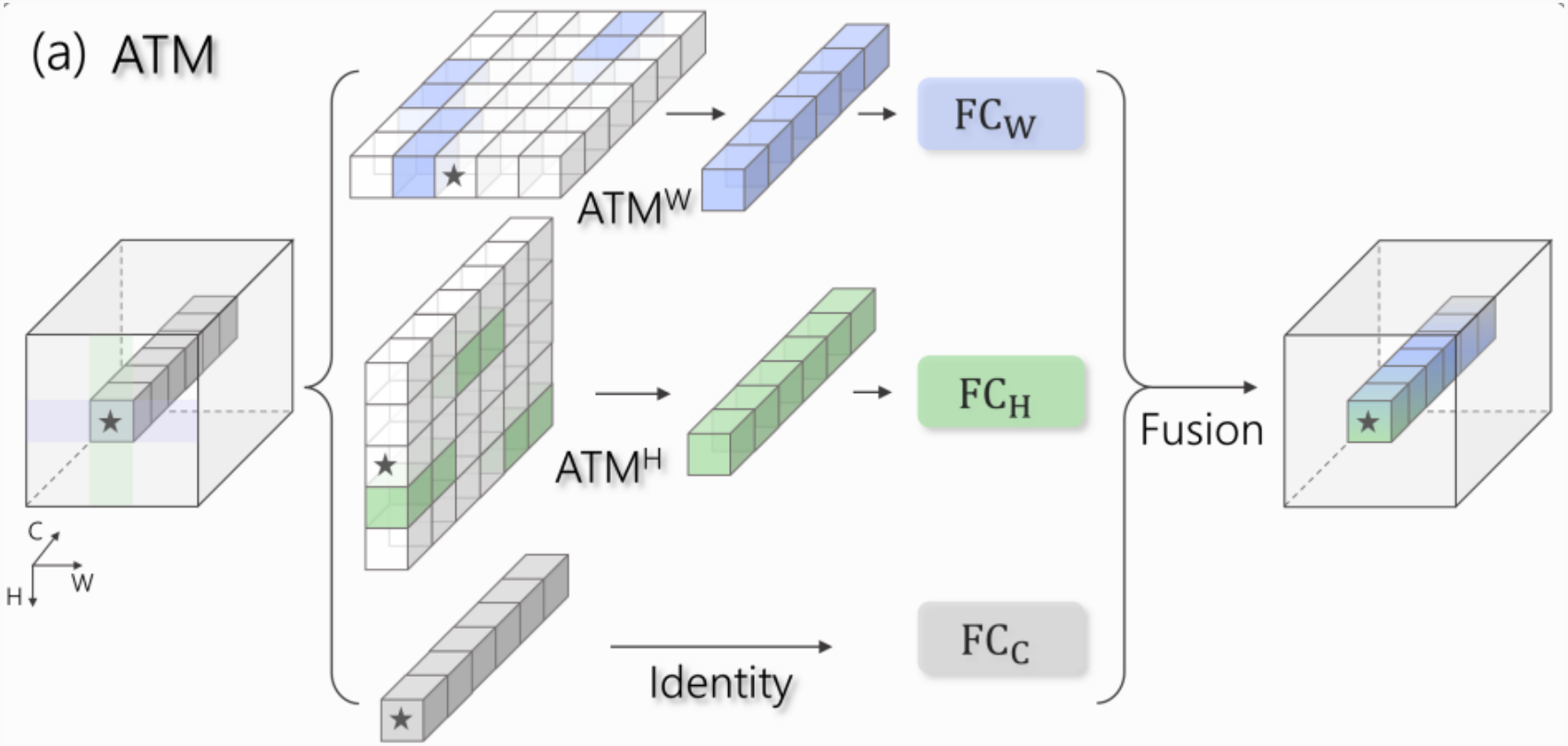
Rysunek 22: AS-MLP. Rysunek z [6].



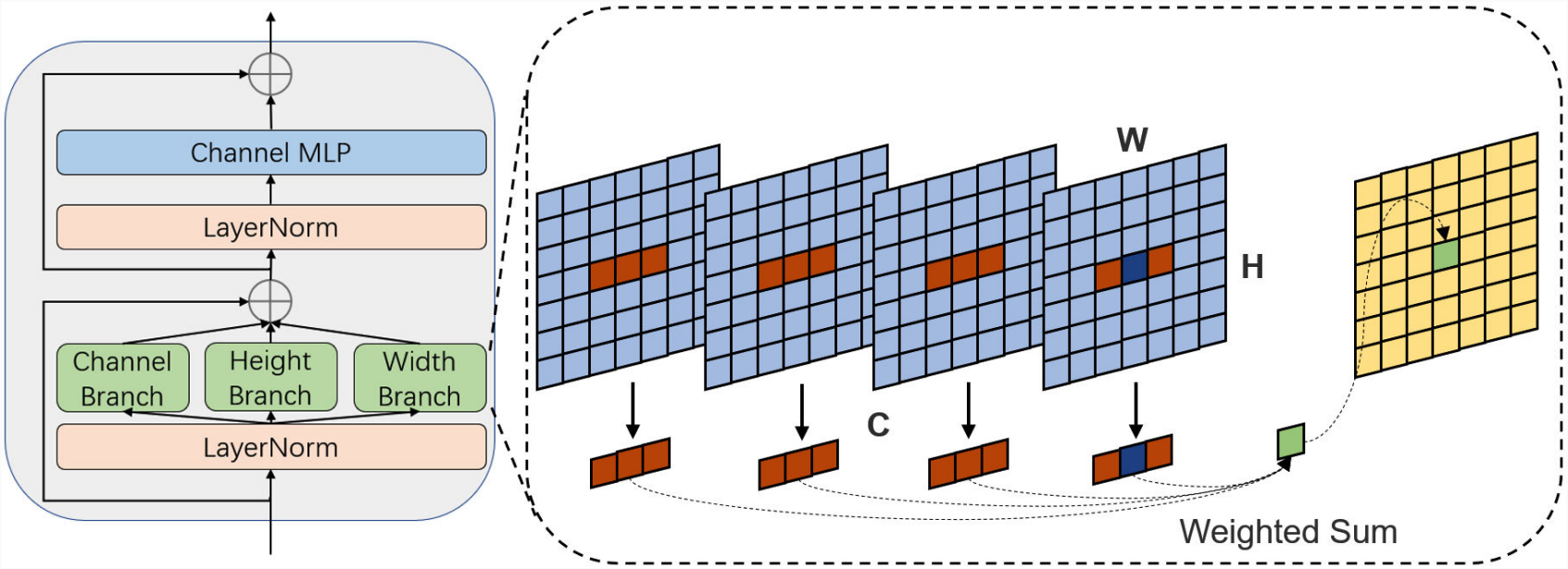
Rysunek 23: AS-MLP. Rysunek z [6].



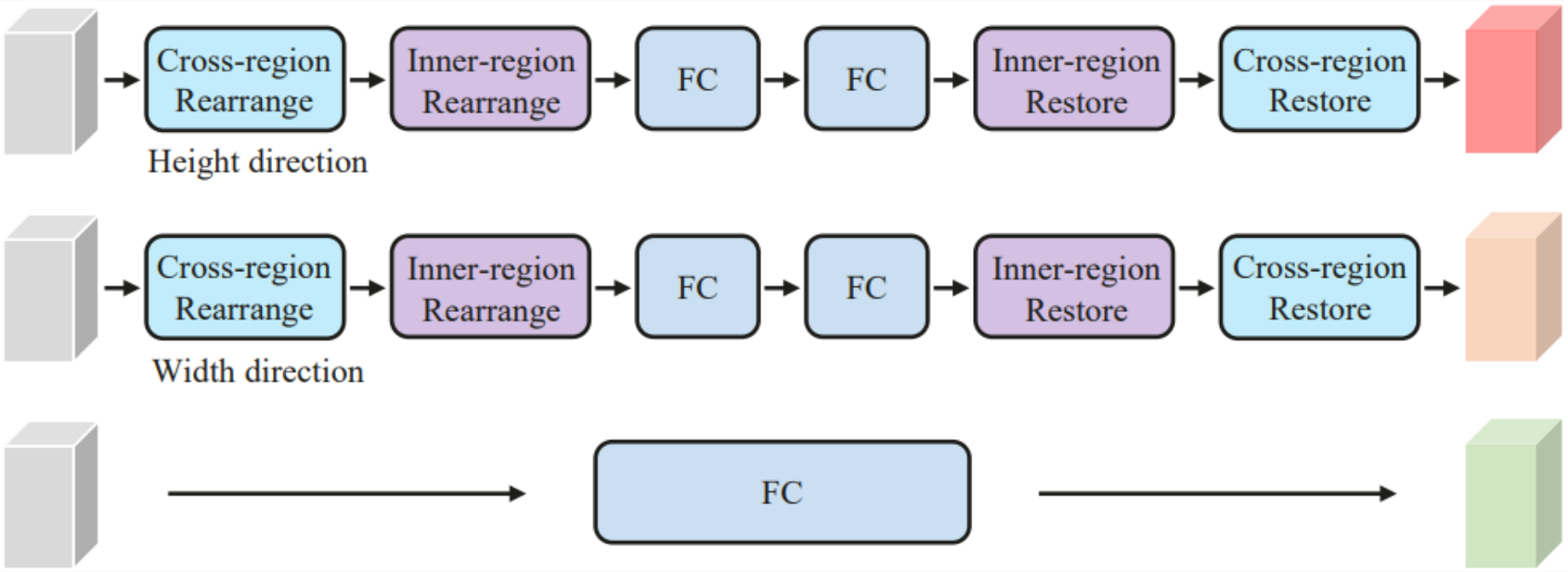
Rysunek 24: MLP with an active token mixer (ActiveMLP) [13]. Rysunek z [7].



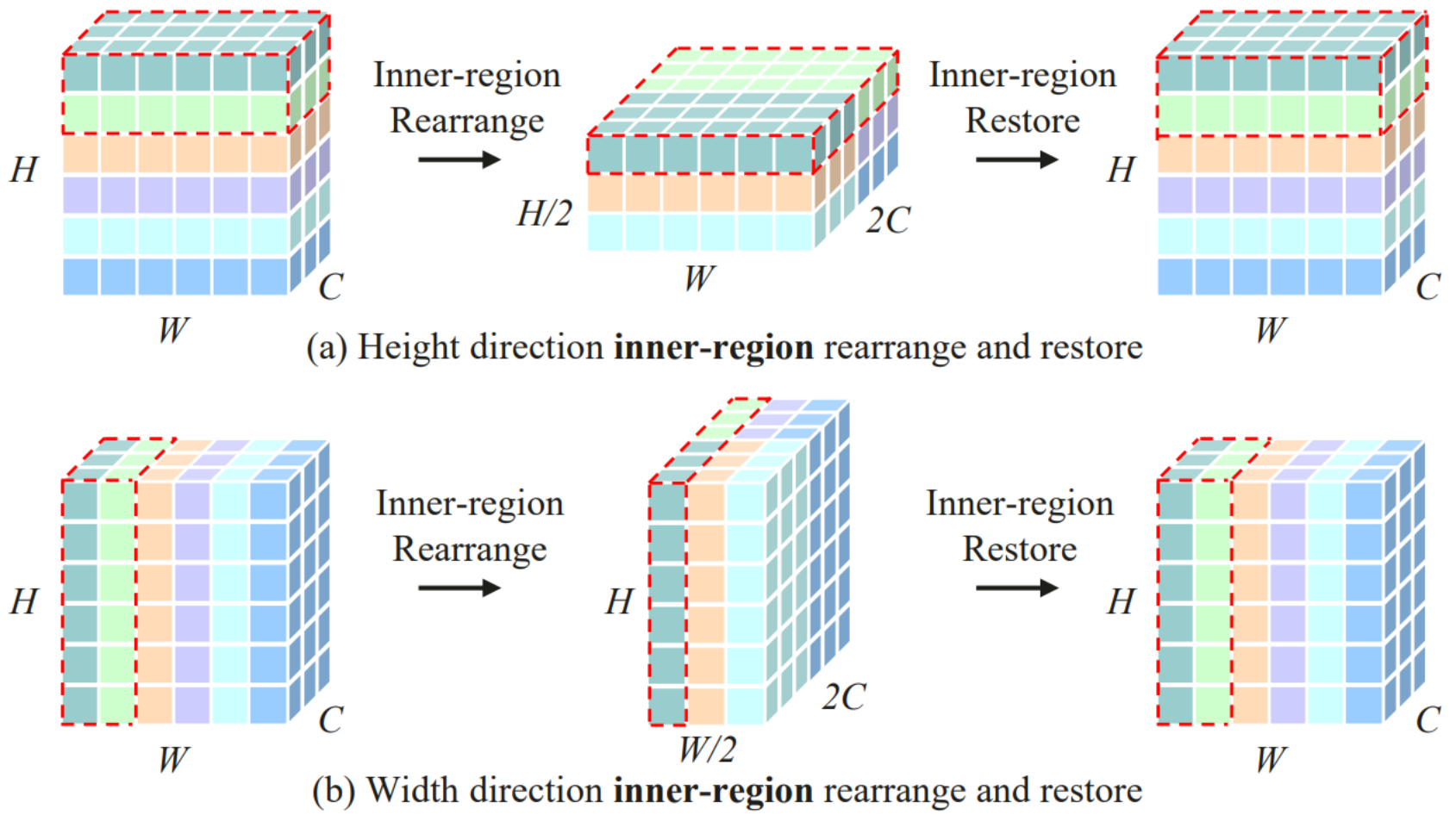
Rysunek 25: ActiveMLP. Rysunek z [13].



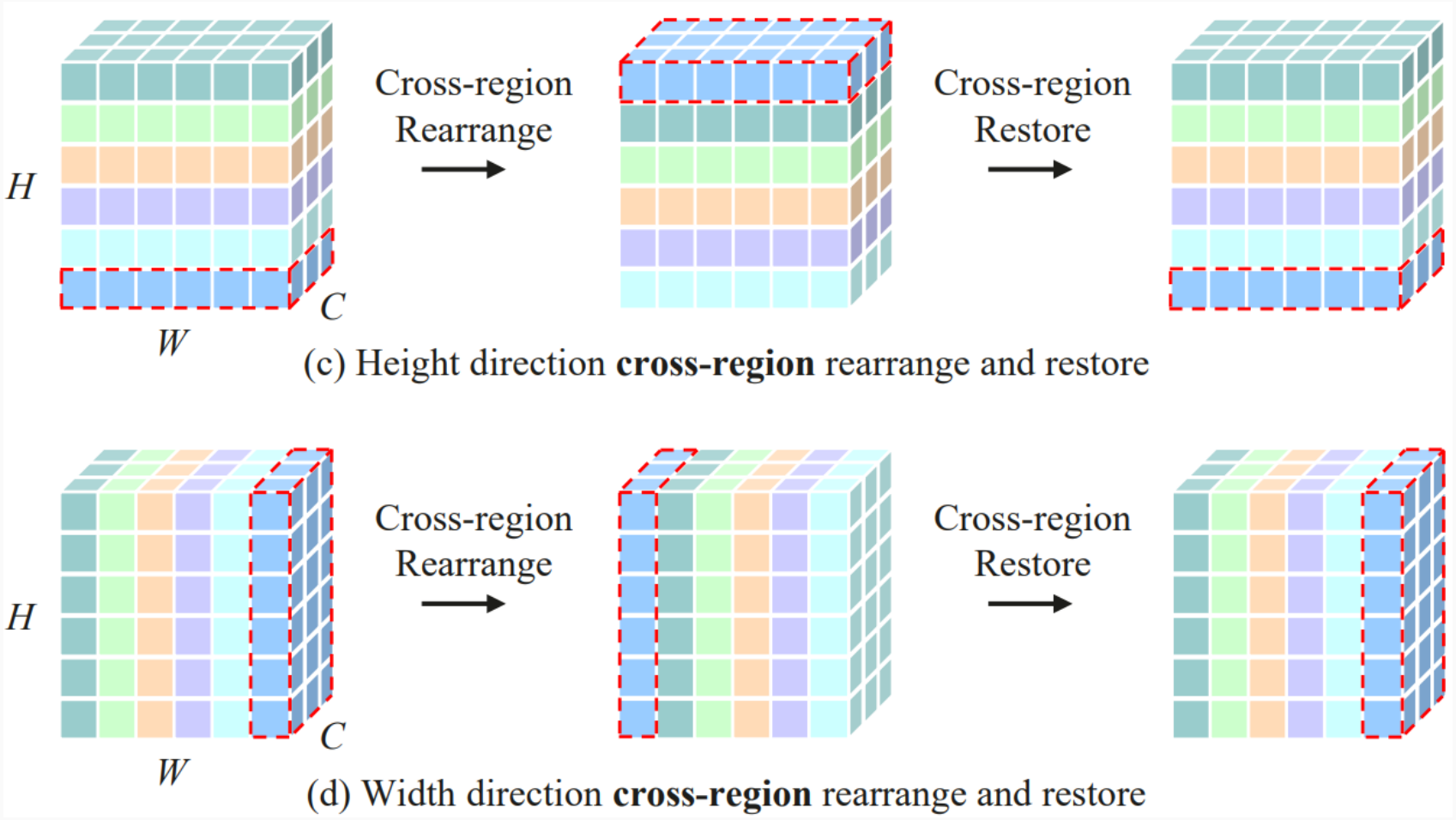
Rysunek 26: MLP with hierarchical rearrangement (HireMLP) [2]. Rysunek z [7].



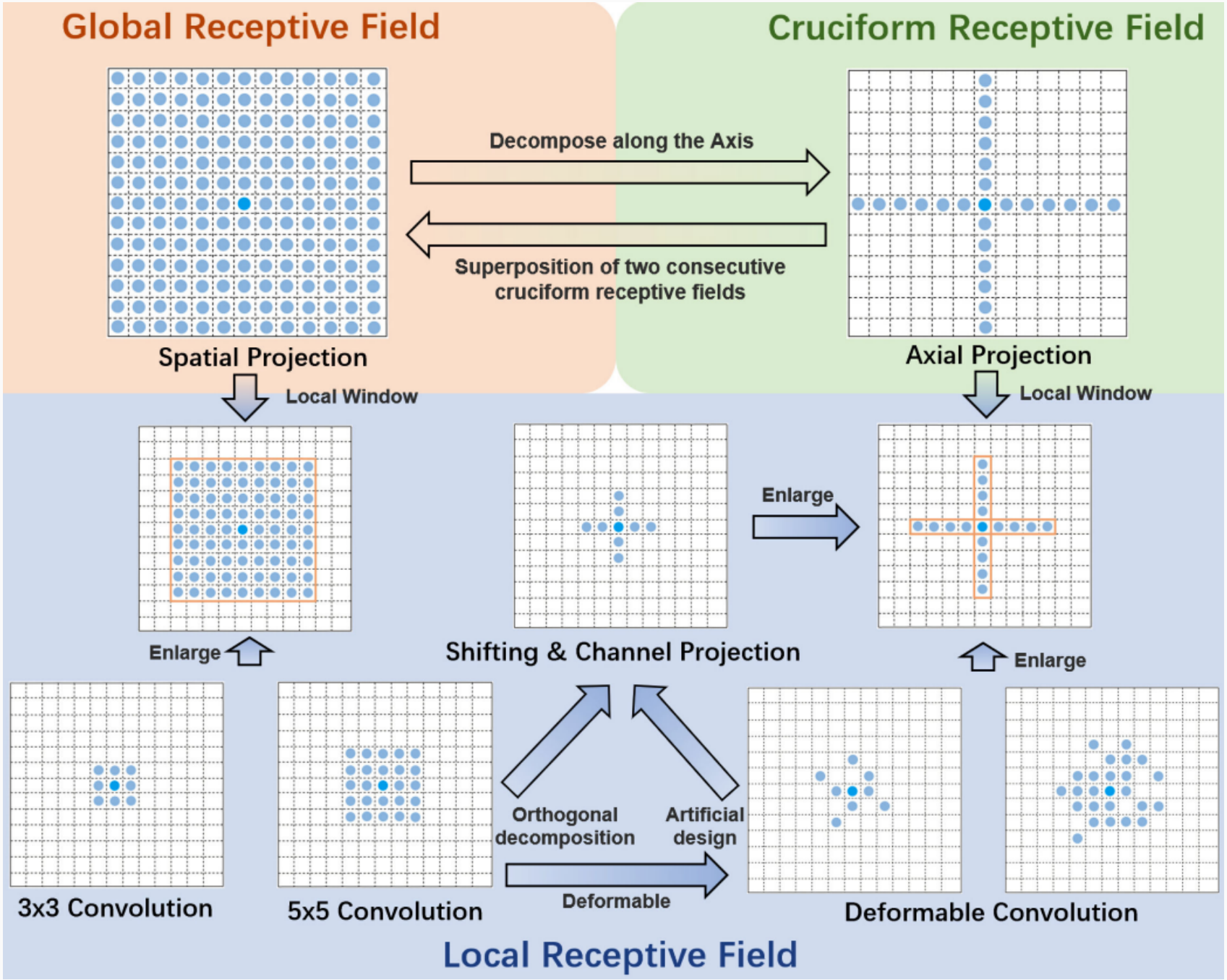
Rysunek 27: HireMLP. Rysunek z [2].



Rysunek 28: HireMLP. Rysunek z [2].



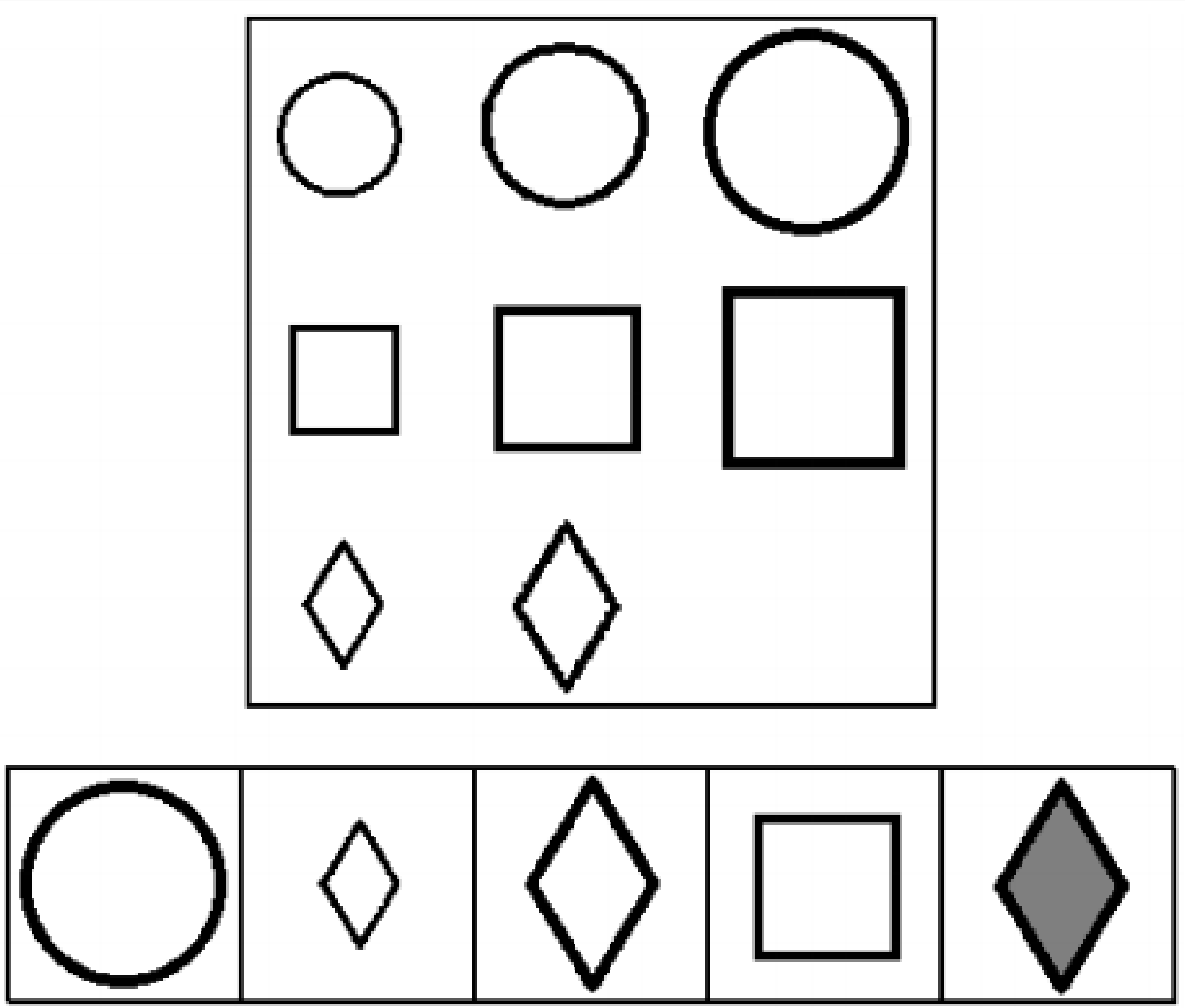
Rysunek 29: HireMLP. Rysunek z [2].



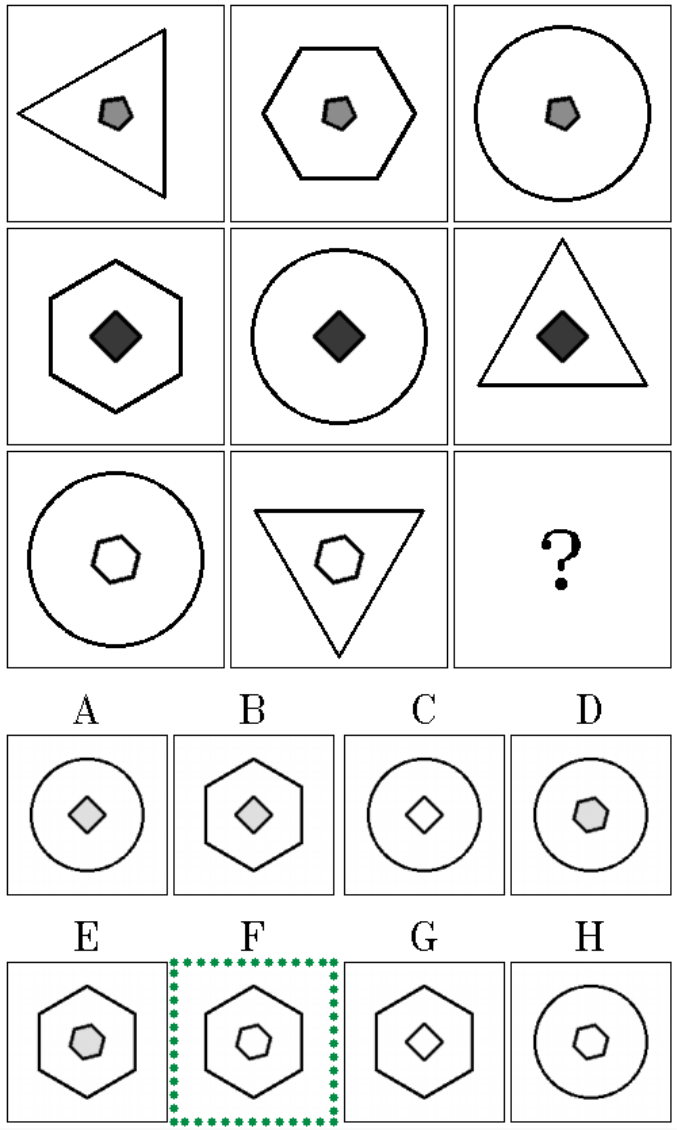
Rysunek 30: Receptive field. Rysunek z [7].

Wyniki: Od Tabeli 4 z https://arxiv.org/pdf/2111.04060.pdf

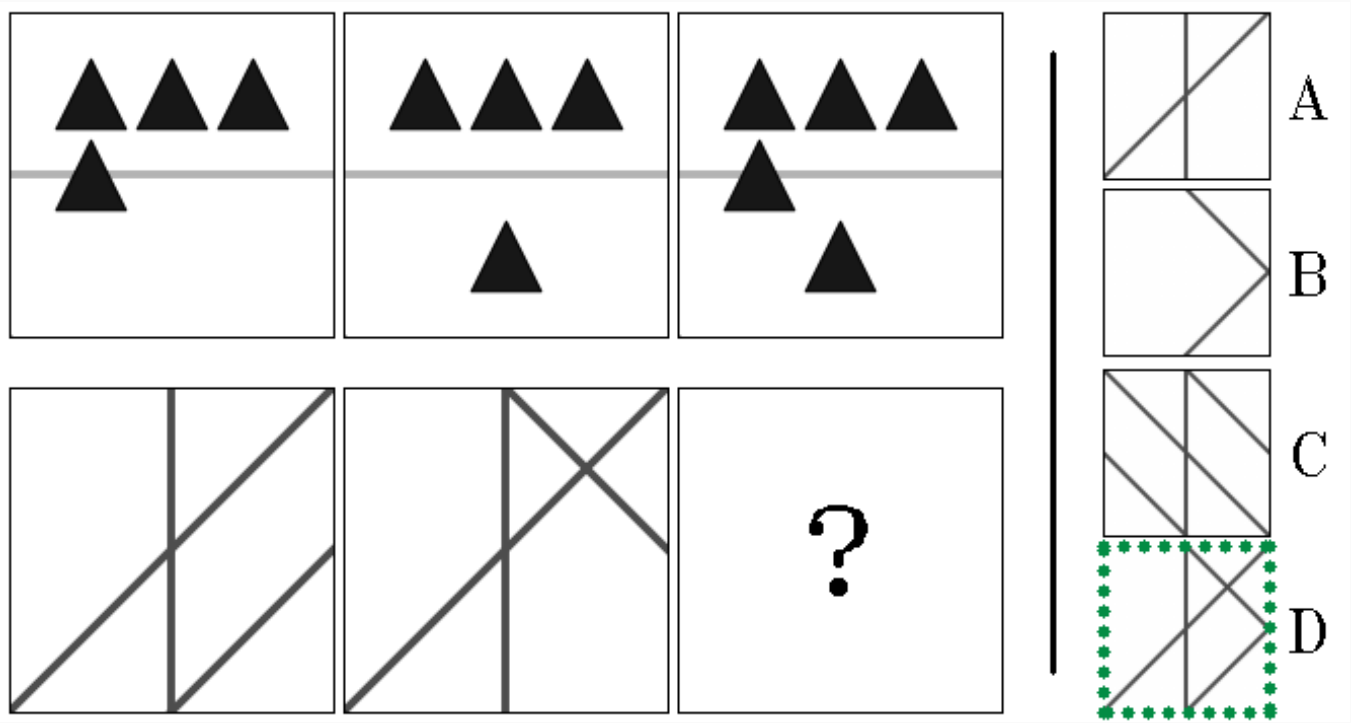
Wykorzystanie modeli do zadań z dziedziny Abstract Visual Reasoning



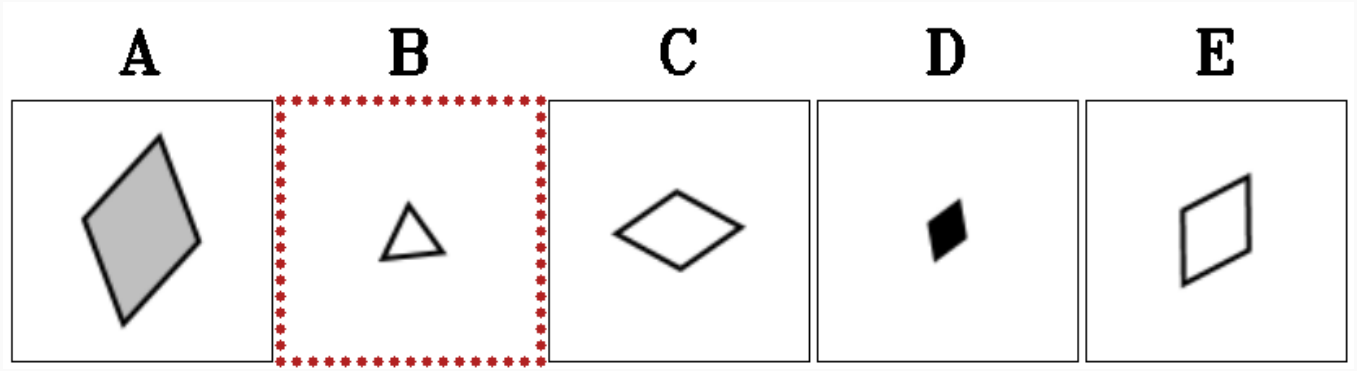
Rysunek 31: Progresywna matryca Ravena (RPM) ze zbioru G-set [9].



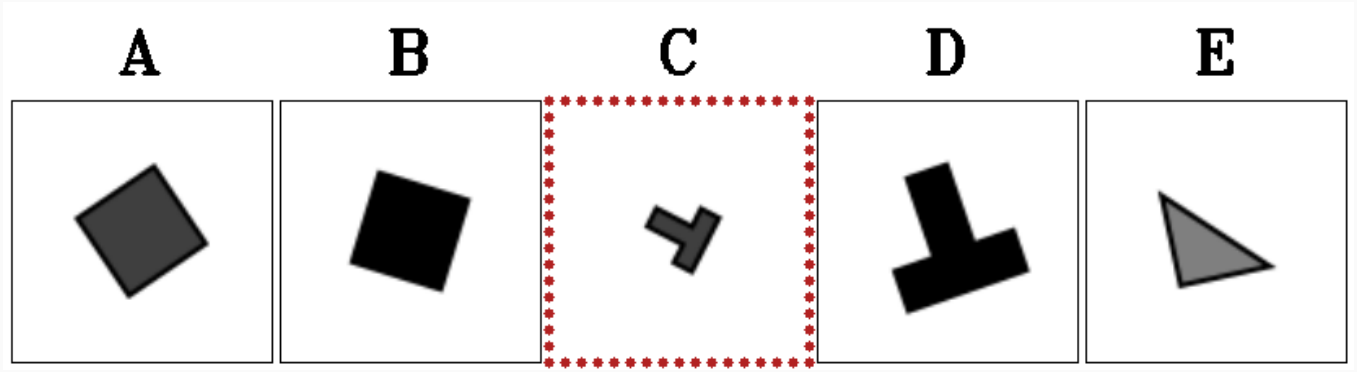
Rysunek 32: RPM z I-RAVEN [5].



Rysunek 33: Problem analogii wizualnej z [3].

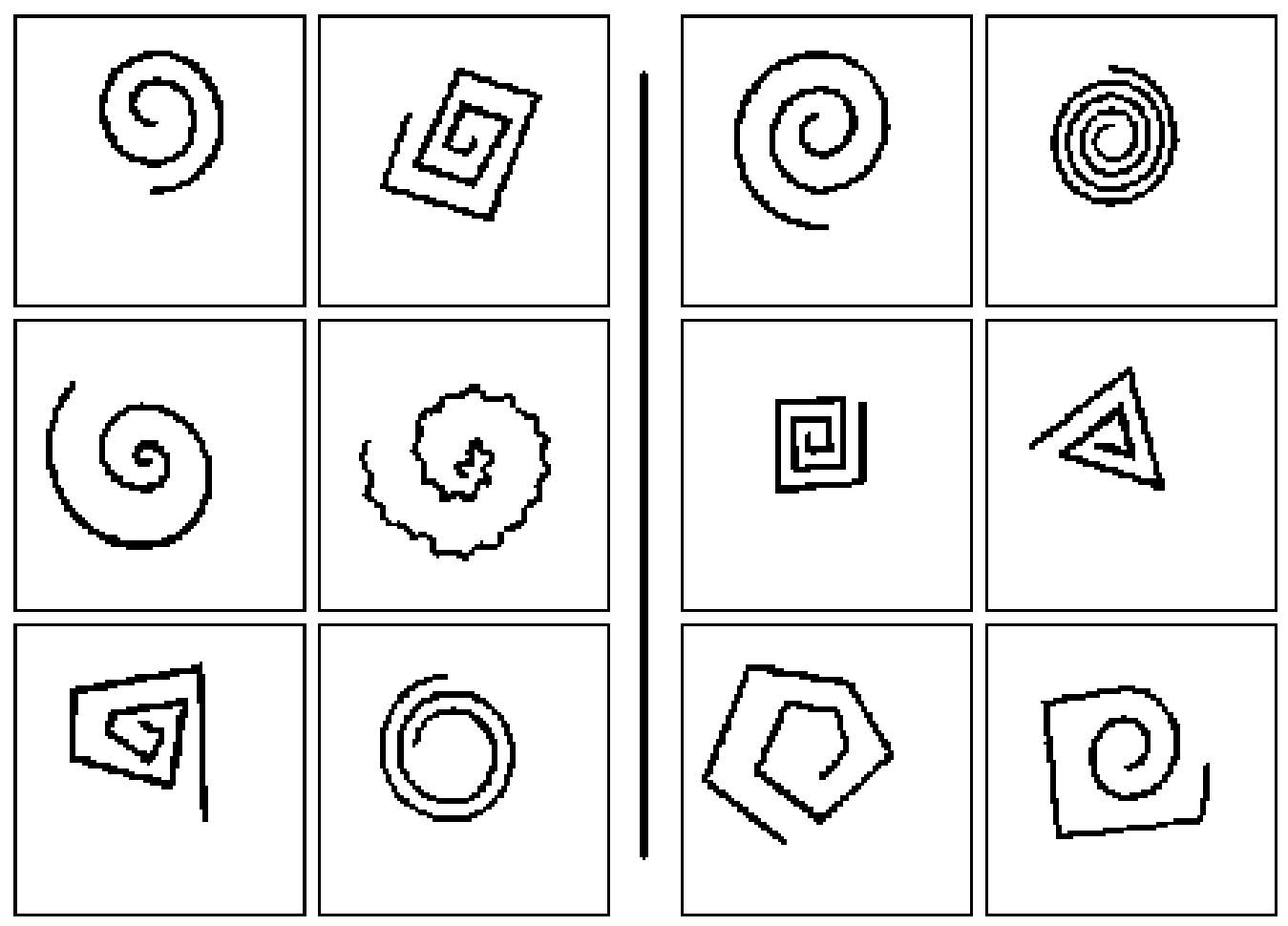


(a)

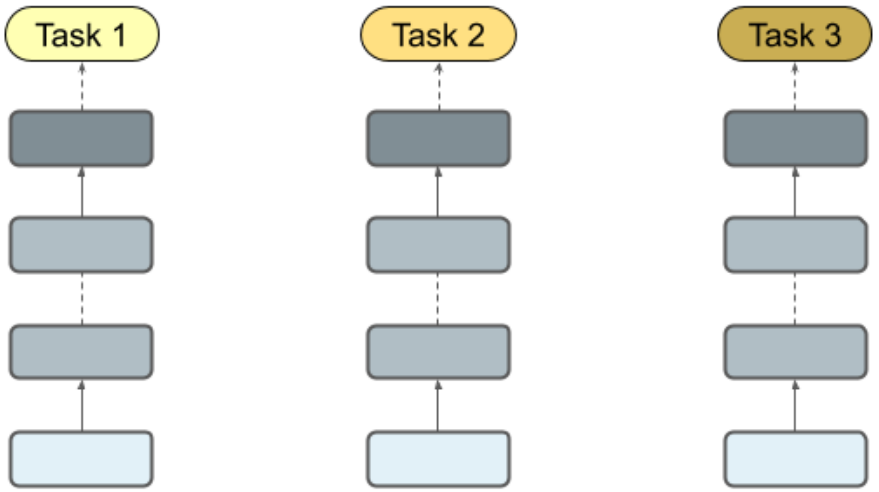


(b)

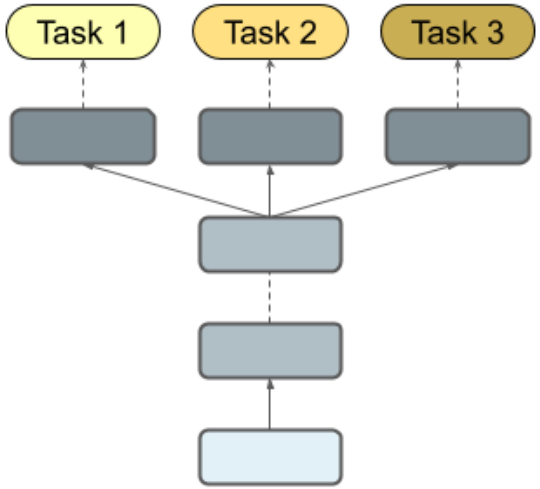
Rysunek 34: Problem odd-one-out z [9].



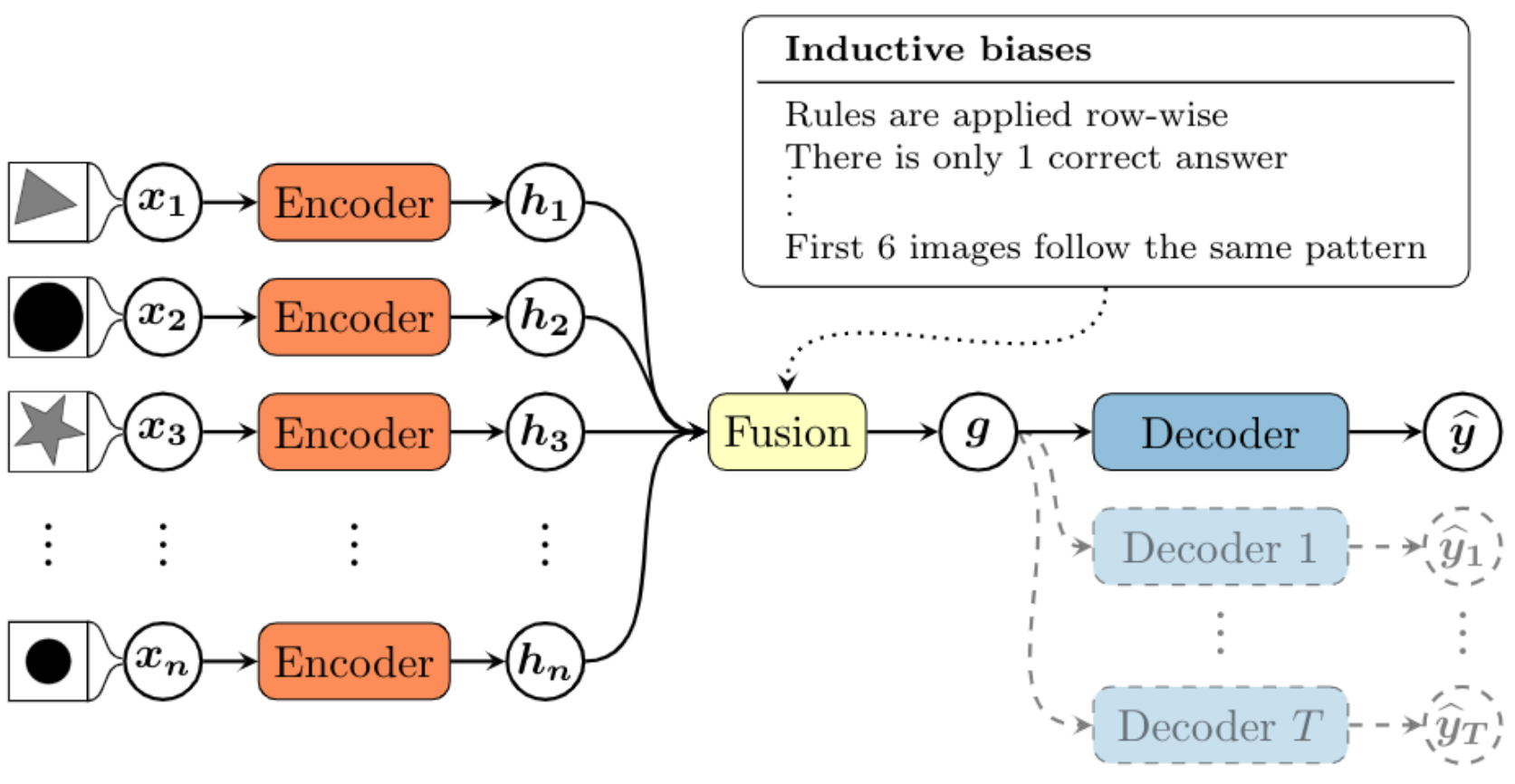
Rysunek 35: Problem Bongarda z [1].



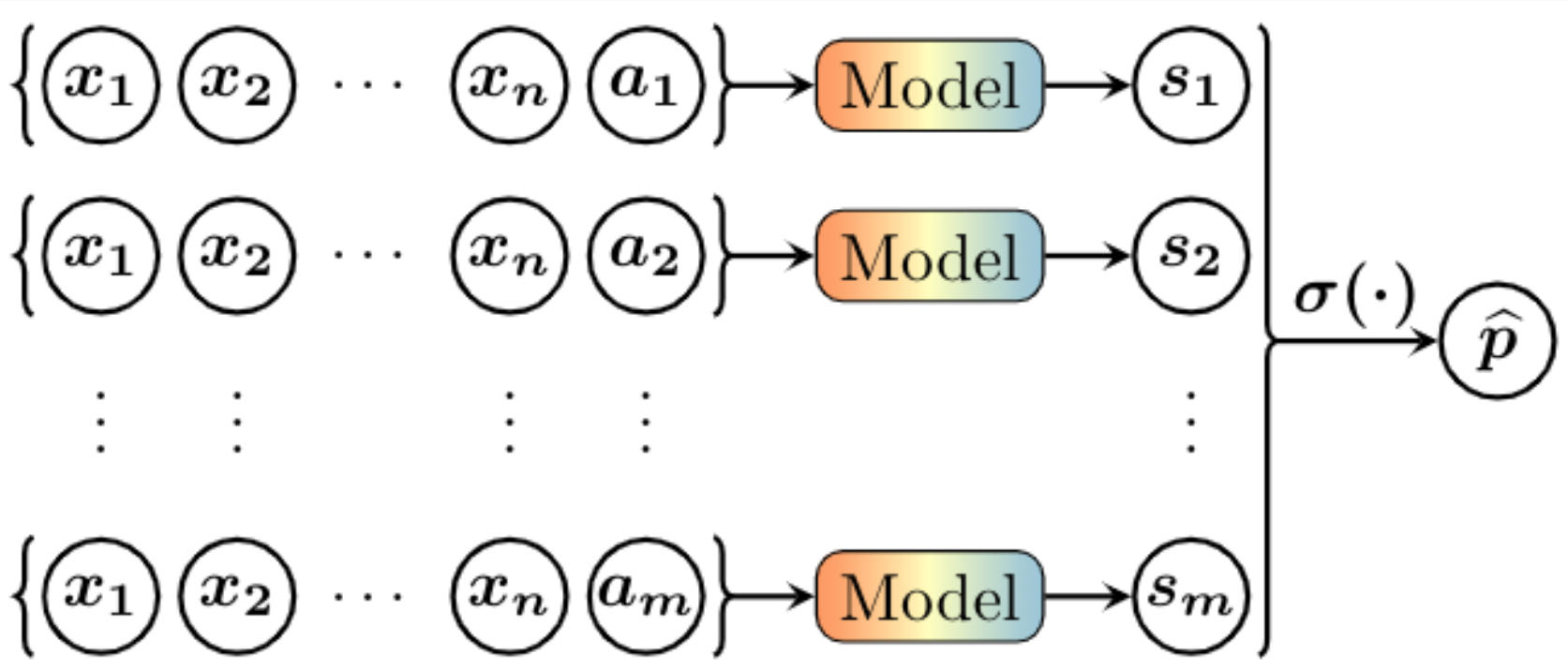
Rysunek 36: Uczenie jedno-zadaniowe (single-task learning). Dla każdego zadania trenowany jest oddzielny model.



Rysunek 37: Uczenie wielo-zadaniowe (multi-task learning). Pojedynczy model jest trenowany do wszystkich zadań jednocześnie.



Rysunek 38: Jednolite spojrzenie na modele do zadań z dziedziny Abstract Visual Reasoning (AVR). Rysunek z [8].



Rysunek 39: Jednolite podejście do rozwiązywania zadań pojedynczego wybory z dziedziny AVR. Rysunek z [8].

Rozłączna perspektywa:

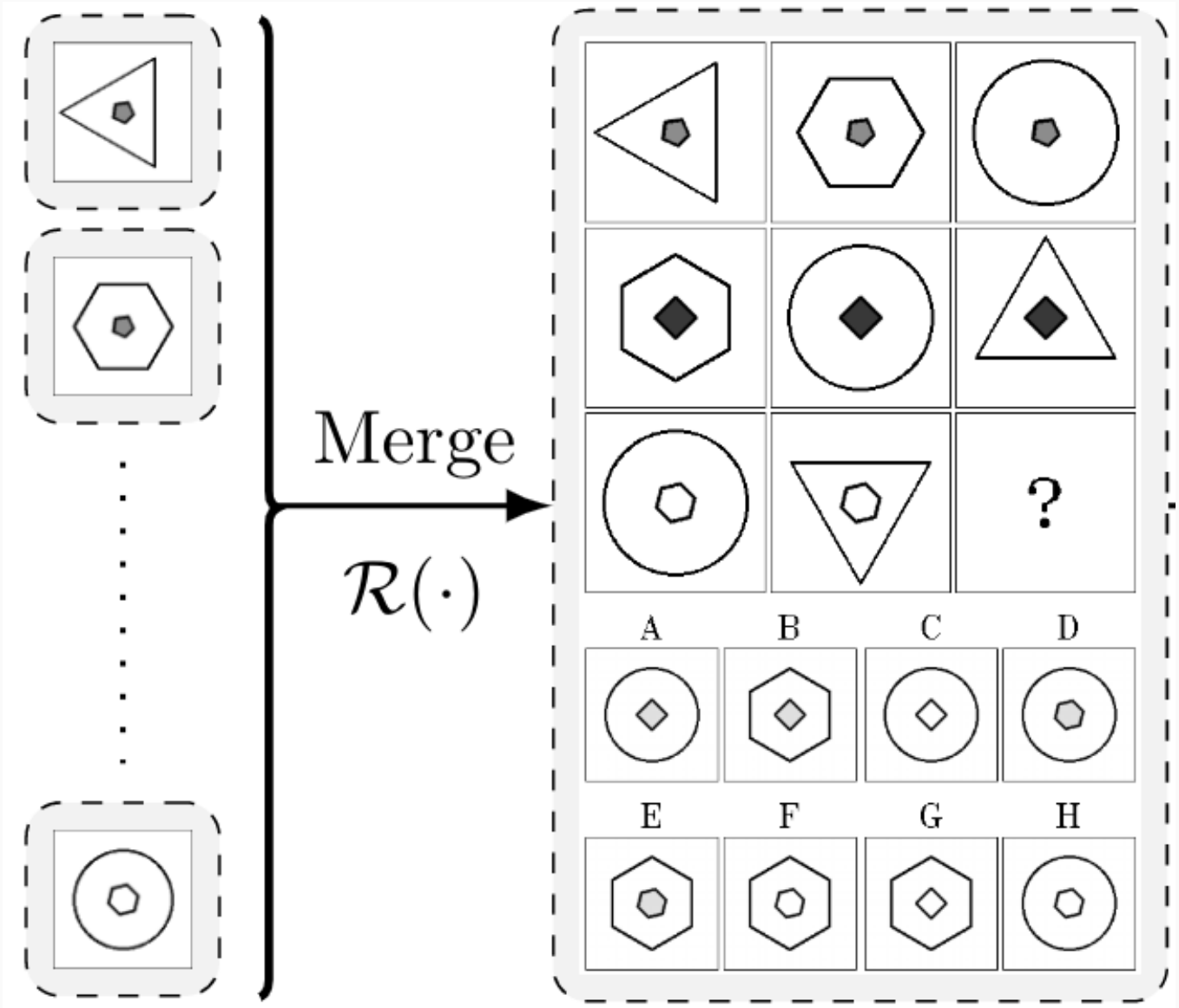
$$t = (\{(\{\mathrm{x}_{i,j}^t\}_{j=1}^{P_t}, \mathrm{y}_i^t)\}_{i=1}^{N_t}, \boldsymbol{\mathcal{S}}^t) \tag{7}$$

Moduł do rysowania metryc:

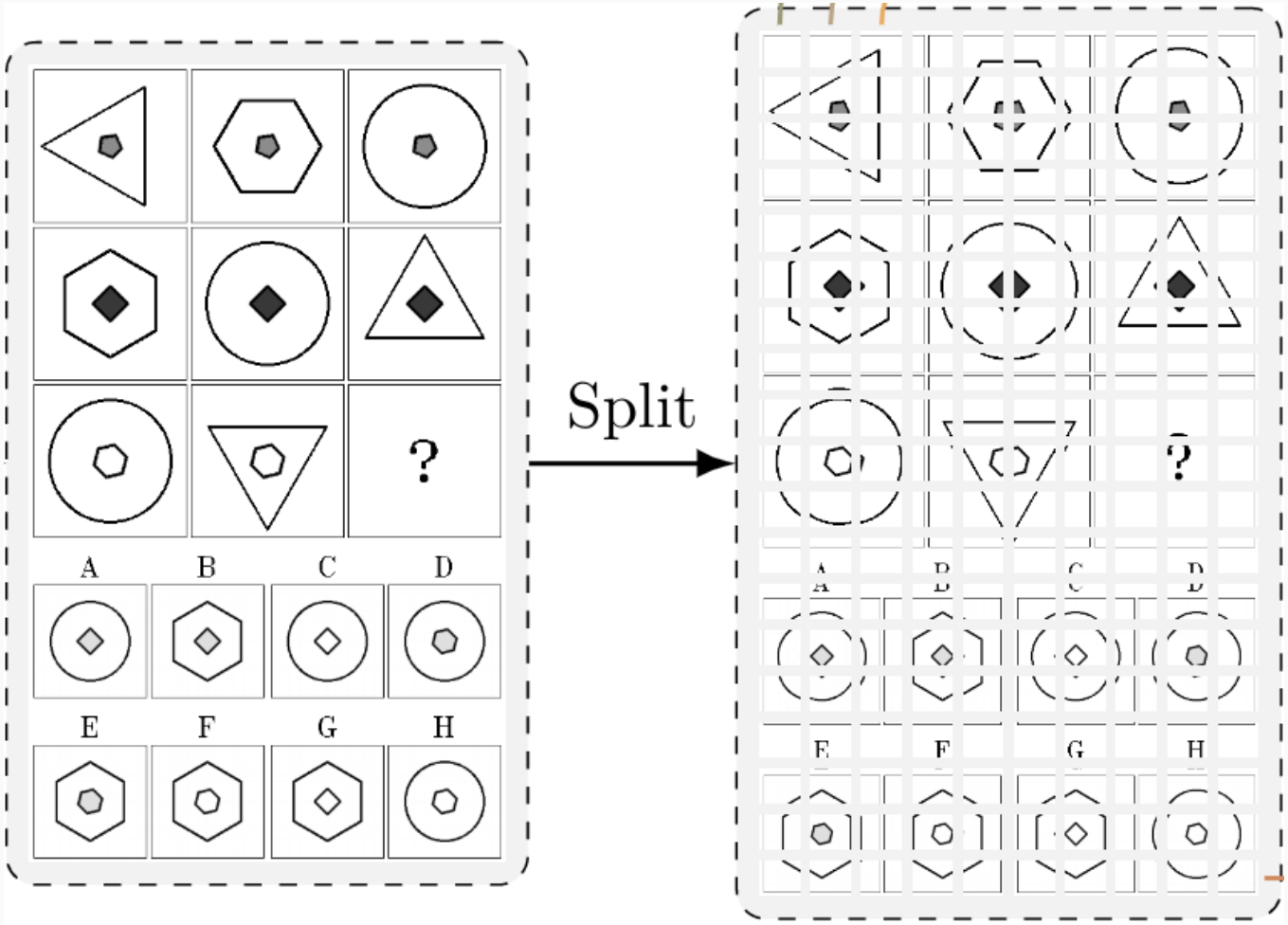
$$\mathcal{R}(\{\mathrm{x}_j^t\}_{j=1}^{P_t} \mid \boldsymbol{\mathcal{S}}^t) = \boldsymbol{\chi}^t \tag{8}$$

Jednolita perspektywa:

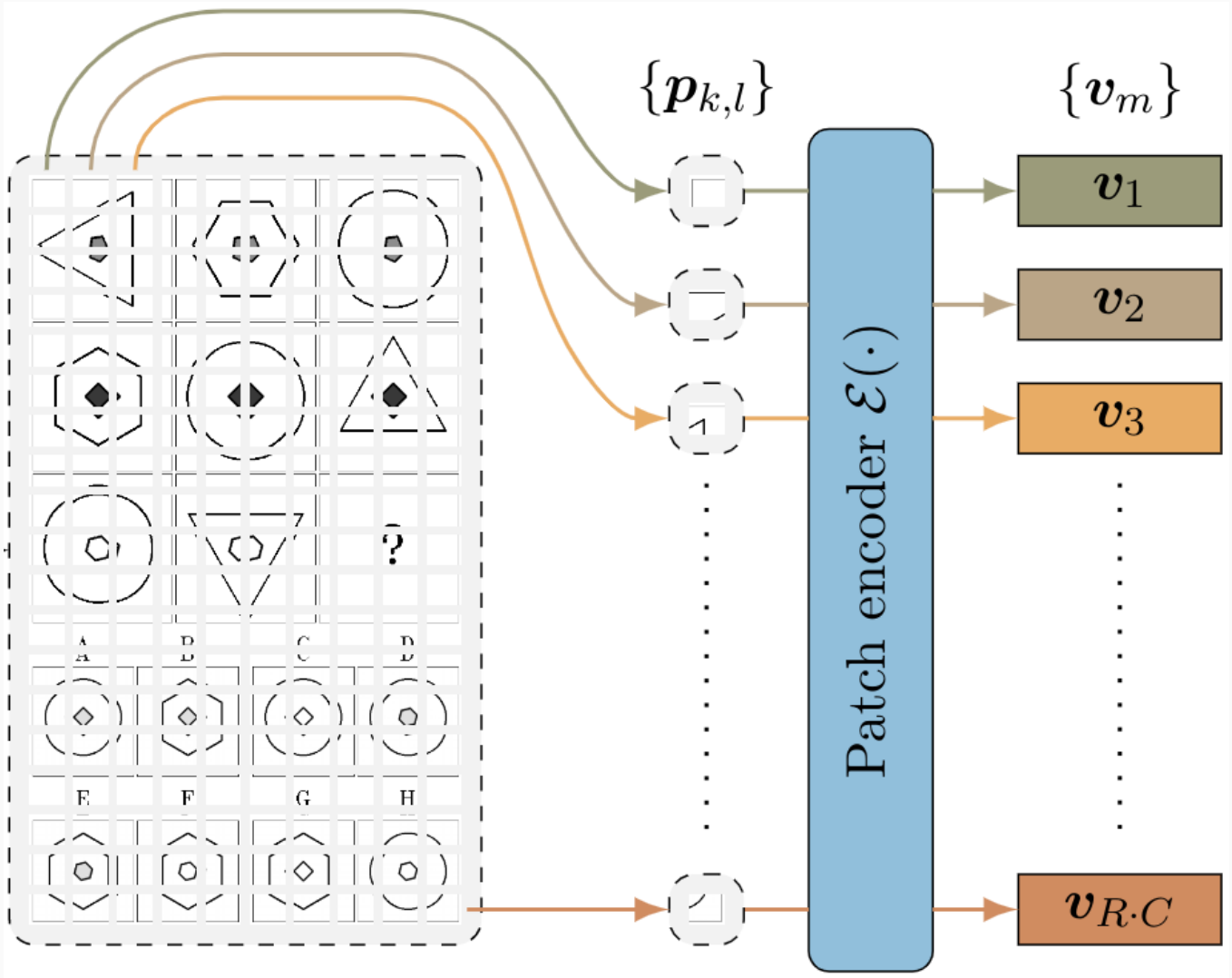
$$t = \{(\boldsymbol{\chi}_i^t, \mathrm{y}_i^t)\}_{i=1}^{N_t} \tag{9}$$



Rysunek 40: Jednolite podejście do rozwiązywania zadań AVR. Krok 1 – Połącz.



Rysunek 41: Jednolite podejście do rozwiązywania zadań AVR. Krok 2 – Rozdziel.



Rysunek 42: Jednolite podejście do rozwiązywania zadań AVR. Krok 3 – Przetwórz.

Eksperymenty:

1. Uczenie jedno-zadaniowe
2. Transfer wiedzy
3. Uczenie wielo-zadaniowe
4. Wykrywanie paneli

Zapraszam do dyskusji

Bibliografia i

Mikhail Moiseevich Bongard.
The recognition problem.
Technical report, Foreign Technology Div Wright-Patterson AFB Ohio, 1968.

Jianyuan Guo, Yehui Tang, Kai Han, Xinghao Chen, Han Wu, Chao Xu, Chang Xu, and Yunhe Wang.
Hire-mlp: Vision mlp via hierarchical rearrangement.
In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 826–836, 2022.

Bibliografia ii


Felix Hill, Adam Santoro, David Barrett, Ari Morcos, and Timothy Lillicrap.
Learning to make analogies by contrasting abstract relational structure.
In *International Conference on Learning Representations*, 2019.


Qibin Hou, Zihang Jiang, Li Yuan, Ming-Ming Cheng, Shuicheng Yan, and Jiashi Feng.
Vision permutator: A permutable mlp-like architecture for visual recognition.
IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022.

Bibliografia iii

Sheng Hu, Yuqing Ma, Xianglong Liu, Yanlu Wei, and Shihao Bai.
Stratified rule-aware network for abstract visual reasoning.
In *AAAI Conference on Artificial Intelligence*, 2021.

Dongze Lian, Zehao Yu, Xing Sun, and Shenghua Gao.
As-mlp: An axial shifted mlp architecture for vision.
arXiv preprint arXiv:2107.08391, 2021.


Ruiyang Liu, Yinghui Li, Linmi Tao, Dun Liang, Shi-Min Hu, and Hai-Tao Zheng.
Are we ready for a new paradigm shift? a survey on visual deep mlp.
arXiv preprint arXiv:2111.04060, 2021.


Mikołaj Małkiński and Jacek Mańdziuk.
A review of emerging research directions in abstract visual reasoning.
arXiv preprint arXiv:2202.10284, 2022.

Bibliografia v

Jacek Mańdziuk and Adam Żychowski.
DeepIQ: A human-inspired AI system for solving IQ test problems.
In *2019 International Joint Conference on Neural Networks*, pages 1–8. IEEE, 2019.

Yuki Tatsunami and Masato Taki.
Raftmlp: Do mlp-based models dream of winning over computer vision?
arXiv e-prints, pages arXiv–2108, 2021.

Bibliografia vi


Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al.
Mlp-mixer: An all-mlp architecture for vision.
Advances in Neural Information Processing Systems, 34:24261–24272, 2021.


Ziyu Wang, Wenhao Jiang, Yiming Zhu, Li Yuan, Yibing Song, and Wei Liu.
Dynamixer: a vision mlp architecture with dynamic mixing.
arXiv preprint arXiv:2201.12083, 2022.


Guoqiang Wei, Zhizheng Zhang, Cuiling Lan, Yan Lu, and Zhibo Chen.
Activemlp: An mlp-like architecture with active token mixer.
arXiv preprint arXiv:2203.06108, 2022.


Tan Yu, Xu Li, Yunfeng Cai, Mingming Sun, and Ping Li.
S2-mlp: Spatial-shift mlp architecture for vision.
In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 297–306, 2022.


David Junhao Zhang, Kunchang Li, Yunpeng Chen, Yali Wang, Shashwat Chandra, Yu Qiao, Luoqi Liu, and Mike Zheng Shou. **Morphmlp: A self-attention free, mlp-like backbone for image and video.**
arXiv preprint arXiv:2111.12527, 2021.