DOI: https://doi.org/10.15276/aait.09.2026.12
UDC 004:83

Metrical loss functions for image segmentation based on convolutional neural networks

Andrii R. Kovtunenko[1)]
ORCID: https://orcid.org/0009-0004-9072-7779; andrii.kovtunenko@nure.ua. Scopus Author ID: 58362751200
Sergii V. Mashtalir[1)]
ORCID: https://orcid.org/0000-0002-0917-6622; sergii.mashtalir@nure.ua. Scopus Author ID: 36183980100
[1)] Kharkiv National University of Radio Electronic, 14, Nauky Ave. Kharkiv, 61166, Ukraine

ABSTRACT

Image segmentation remains a fundamental challenge in computer vision, with neural network training heavily dependent on appropriate loss functions. While common losses such as Dice are widely used, they lack rigorous mathematical foundations as proper distance metrics and do not fully capture the geometric structure of partitions. **We introduce a weighted metric** for comparing segmentation based on partition theory that satisfies all metric axioms, including the triangle inequality. The proposed metric compares partitions through symmetric difference and intersection operations, incorporating both spatial structure and semantic features via a weight function characterizing region properties, such as color, texture and other. **We prove that the** proposed functional forms a proper metric space on weighted partitions under specified conditions, with particular emphasis on establishing the triangle inequality. **Experimental verification** on synthetic segmentation tasks demonstrates feasibility, although practical implementation faces challenges, such as the need for differentiated segment extraction, which can be solved using the Straight-Through Estimator approximation. The triangle inequality property opens up opportunities for hierarchical approaches to segmentation and efficient partition search. This work bridges the gap between geometric clustering theory and deep learning-based segmentation, providing a theoretically grounded alternative to heuristic loss functions and also experimentally proves the possibility of using the proposed metric as a loss function when training convolutional neural networks.

Keywords: Image segmentation; loss functions; convolutional neural networks; deep learning; metric; partition; computer vision

For citation: **Kovtunenko A. R., Mashtalir S. V. "Metrical loss functions for image segmentation based on convolutional neural networks". *Applied Aspects of Information Technology.* 2026; Vol. 9 No. 2: 172–184. DOI: https://doi.org/10.15276/aait.09.2026.12**

INTRODUCTION

Image understanding mostly depends on segmentation, as separating an image into several disjoint (or weakly intersecting) regions whose characteristics, such as intensity, color, texture, shape, etc., are similar. Segmentation algorithms have been widely investigated. However, image content formal descriptions using only low-level features extracted from each region do not always provide spatial reasoning; i.e., achieving a totally correct segmentation may result in under-segmentation, over-segmentation, missed regions, and outlier regions. The last decade has determined a promising direction that bridges a semantic gap between low-level features and human concepts and consists of convolutional neural networks (CNN) architecture development.

Architectures for solving computer vision problems have come a long way from simple full convolutional models to transformer-based models. Fully convolutional network (FCN) [1] allowed solving the segmentation for images of arbitrary resolution, without the need to post-process the results. Then the U-Net [2] architecture, which greatly improved the processing of fine details and boundary search by introducing skip connections, which allowed combining low-level spatial features with high-level features that are responsible for semantics. The SegNet [3] model proposed a more efficient approach to processing high-resolution images. The architecture also used an encoder-decoder structure, but with modified upsampling in the decoder, using stored max pooling indices instead of transmitting full feature maps. Approaches for instance segmentation evolved in the same way. Mask R-CNN [4], based on Faster

R-CNN [5], and introduced a two-stage algorithm: region of interest (RoI) prediction followed by parallel mask generation for each object and classification. This approach became the standard for a long time, providing accurate masks even for overlapping objects, but it required significant computational resources. The Gated-SCNN [6] method proposed an architecture consisting of two branches. The main branch extracts features, which are subsumed into a second branch that works with the computed gradients of the original image through a canny filter via a Gated Convolution Layer (conceptually equivalent to modern attention mechanisms). The results

of these two branches are then combined via ASPP to form the final segmentation result. The key contribution of this model was dual task loss, which compares the predicted boundaries with the true boundaries. The next FastFCN [7] method optimized computational efficiency using Joint Pyramid Upsampling (JPU), where input feature maps are processed by separable convolutions with different dilation rates.

Let's conclude with an overview of models that utilize transformers internally. MaskFormer [8] changed the approach to generating feature masks from pixel-based classification to initially generating a set of masks and then classifying them. Backbone extracts embeddings from which binary object masks are generated after the pixels decoder and transformer decoder processing. Mask2Former [9] further developed the ideas of MaskFormer by replacing cross-attention with masked-attention and adding feature maps of different resolutions to the decoder. Masked attention allowed focusing attention around objects of interest, speeding up convergence.

As architectures evolved, loss functions also underwent their own evolution, as the choice of loss function is an important aspect for efficient problem solving and model training. Loss function is a function that is designed to calculate the error between the predicted values generated by the model and the reference values of the target variable. The main goal of the machine learning model training process is to minimize the value of the loss function by iteratively adjusting the model parameters. For computer vision tasks, let us distinguish the following subclasses for loss functions: for classification, for segmentation, for detection, and for metric learning. Let us start with the Distribution-based loss functions for classification, which are used in tasks where it is necessary to assign an image or its parts to predetermined classes. These functions estimate the discrepancy between the predicted probability distribution of the classes and the true value. Table 1 presents a comparison of distribution-based loss functions based on key characteristics that affect their applicability in different scenarios. Sensitivity to extremes indicates how strongly the loss function responds to examples with high probability or value, which affects the overall learning dynamics.

Cross-entropy loss is a fundamental loss function for classification tasks. It measures the discrepancy between two probability distributions. Despite its widespread use, Cross-entropy faces problems when dealing with unbalanced datasets, which has led to the development of its modifications. Balanced CE introduces coefficients that take into account the frequency of occurrence of each class in the dataset. Weighted CE, in turn, extends this concept by introducing weights for different classes regardless of their frequency. This provides additional flexibility for specific tasks. Focal loss allows dealing with highly dispersed datasets, and reduces the contribution of correct answers with high probability (easy examples), allowing the model to respond more to incorrect answers than to correct ones.

Table 1. **Comparison of distribution-based loss functions**

Name	Imb Res.	Diff.	Compl.	Sens. Extr.	Params
Cross-Entropy	Low	Yes	Low	High	No
Balanced CE	High	Yes	Low	Medium	No
Weighted CE	High	Yes	Low	Medium	Yes
Focal Loss	Very high	Yes	Medium	Controlled (γ)	Yes

Source: **compiled by the authors**

In this table, the following abbreviations are used: Imb. Res. – resistance to class imbalance; Diff. – differentiability; Compl. – computational complexity; Sens. Extr. – sensitivity to extreme values; Params – presence of additional hyperparameters.

Table 2. **Comparison of region-based loss functions**

Name	Imb. Res.	Diff.	Compl.	Sens. Extr.	FP/FN Ctrl.
Dice Loss	High	Yes	Low	Medium	Condi-tional
Tversky Loss	Very high	Yes	Low	Controlled (α/β)	Yes
IoU Loss	Medium	Partial	Medium	Medium	No
Lovász-Softmax	Medium	Yes	High	High	Yes

Source: **compiled by the authors**

Abbreviations used: Imb. Res. – resistance to class imbalance; Diff. – differentiability; Compl. – computational complexity; Sens. Extr. – sensitivity to extreme values; FP/FN Ctrl. – control over false positives and false negatives.

Unlike classification, which estimates whether the entire image belongs to a particular class, segmentation requires pixel-by-pixel prediction of class membership. Consider functions for solving

such problems. Loss functions for segmentation can be categorized into two main subtypes: region-based – estimating region correspondence; and boundary-based – focusing on the accuracy of object boundary detection.

Dice Loss is designed to compare the dice coefficient (also known as F1-measure) between two sets – the resulting segmentation mask and the reference set. The gradients of this function are unstable, and gradient explosion is also possible. The instability and potential explosion of gradients are caused by the Dice Loss denominator approaching zero when the target region is minimal or absent. This leads to unbounded derivative calculations, which occur primarily when the total pixel area of both the predicted and true masks is extremely small. A combined Log-Cosh Dice Loss function [10] was proposed to overcome the disadvantages of non-smoothness (non-convexity) and gradient problems. The Tversky Loss [11] function is a generalization of Dice Loss. It adds the ability to control the balance between false positives (FP) and false negatives (FN) errors. It is also worth noting that it can be unstable in the initial training phase.

The next function, IoU, is a common metric for evaluating the quality of object detection, which can also be used as a loss function for the segmentation task – IoU Loss (Jaccard loss), but the function is discontinuous. A differentiable approximation method must be applied for differentiation. e.g., a smoothing parameter ò is added or a log-sum-exp trick is used to improve numerical stability. To overcome this drawback, various modifications have been proposed, such as Generalized IoU (GIoU) [12] Distance IoU (DIoU) [13], Complete IoU (CIoU) [14], Soft IoU [15]. Moreover, Lovász-Softmax Loss [16] also optimizes IoU but in another manner.

While region-based features focus on evaluating the concurrence of segmentation regions, boundary-based features focus on the accuracy of object boundaries. This is especially important in medical tasks where accurate object boundary detection can be critical. The comparison is summarized in Table 3.

Boundary Loss [17] focuses exclusively on contours and is often used in combination with other loss functions, such as Dice Loss, to achieve better results in medical image segmentation tasks.

Hausdorff Loss [18] is based on the Hausdorff distance and measures the maximum distance between the border points of the predicted and true masks. Since the Hausdorff distance is not differentiable, it requires the creation of different approximations. A special feature of this function is that it penalizes even for single boundary deviations, which can be a very important criterion for solving the problems at hand. We have considered functions for segmentation, but they can also be applied to solving object detection problems. In this case, the object areas are those pixels that fall within the bounding boxes.

Table 3. **Comparison of boundary-based loss functions**

Name	Imb. Res.	Diff.	Comp.	Bound.	Noise	Size Inv.
Boundary Loss	High	Partial	High	Medium	High	Medium
Hausdorff Loss	Medium	No	Very high	High	Low	Low
Active Contour	High	Yes	Medium	Low	High	High

Source: **compiled by the authors**

Abbreviations used: Imb. Res. – resistance to class imbalance; Diff. – differentiability; Comp. – computational complexity; Bound. – sensitivity to boundaries; Noise – noise immunity; Size Inv. – size invariance.

Let us consider one more type of functions -- embedding-based losses. These are loss functions that work with vector representations (embeddings) of objects, placing them in a multidimensional space in such a way that semantically close objects are closer to each other and different objects are farther away. Usually applied in representation learning. The representation learning approach is to extract useful features from the data or to transform it into another feature space, where certain characteristics of the data become more explicit and can be used for other tasks. A comparison of some features is presented in Table 4.

Table 4. **Comparison of embedding-based loss functions**

Name	Imb. Res.	Diff.	Compl.	Sens. Extr.	Params
Triplet Loss	Medium	Yes	High	Average	Yes
Center Loss	High	Yes	Medium	High	Yes

Source: **compiled by the authors**

Abbreviations used: Imb. Res. – resistance to class imbalance; Diff. – differentiability; Compl. – computational complexity; Sens. Extr. – sensitivity to extreme values; Params – indicates whether the loss includes tunable parameters.

Triplet loss [19] is a loss function that teaches the network to form a feature space in which objects of the same class are closer together and objects of different classes are farther apart. The learning is based on triplets. A triplet is a set of three elements: an anchor; a positive example (positive) of the same class as the anchor; a negative example (negative) from a class that does not belong to the anchor class. The function minimizes the distance between the anchor and the positive example while maximizing the distance between the anchor and the negative example, thus creating a space in which classes are well distinguishable. Center Loss [20] – is a regularization technique aimed at generating significantly spaced representations of classes in the feature space. The main goal of this approach is to maximize the interclass distance. Center Loss complements the classical teacher learning loss function by introducing an additional penalty component, the value of which is proportional to the distance between individual samples of a class and the corresponding center of this class in the feature space.

Having considered the main types of loss functions for various computer vision tasks, it is important to note that combined loss functions are often used in practice. They combine the advantages of different approaches, compensating for the individual disadvantages of separate functions. Combined functions are usually a weighted sum of several loss functions, where coefficients can be fixed or adjustable parameters.

2. GROUNDWORK AND PROBLEM STATEMENT

Region-based loss functions are suitable for various segmentation approaches. But there exists a demand to bridge a semantic gap between low-level features and human concepts. Consideration of the spatial image content of the collective structure of region families is an advance over single-region analysis. Thus, there are many reasons for the study of set partitions matching since they are models of arbitrary clustering. In the presence of multiple segmentations (comparison of different algorithms, search for a compromise within the range from an under- to over-segmentations, learning for semantic, instance, panoptic segmentations), there appears necessity to take into consideration metrical properties of quotient set comparisons which due to consolidated geometrical content will not be sensitive to varying acquisition, partial objects occlusions, colors transformations also. The transition from the measure of similarity (satisfying reflexivity and symmetry) to the analysis of metrics (in addition, the triangle inequality is valid) means the possibility of preparatory processing providing a significant increase in performance. Pairwise comparison is generally more powerful than exhaustive search when multiple comparative estimates (using the reverse triangle inequality in retrieval algorithms) enable quickly eliminating dissimilar elements.

There have been many efforts on clustering, comparing, and matching two different partitions of different sets that reappear continually. However, they do not always pay attention to the triangle inequality. The last one is important at least for controlled learning of networks, when it is effective to take advantage of the fragmentation similarity degree, e.g., from very different partitions to substantially similar ones at different stages of learning.

Among the most well-known metrics should be highlighted van Dongen [21], Larsen [22], Mirkin [23], variation of information (a.k.a. Meila metric) [24], the Earth Mover's Distance (EMD) [25] metrics. However, most of them are valid only for finite-dimensional sets and either have considerable computational complexity or have low sensitivity under meaning changes of partitions.

Let U be an arbitrary finite set with cardinality N. Crisp clustering generates different partitions $X = \{[x]_1, [x]_2, \dots, [x]_m\}$ (where $[x]_i \neq \varnothing$, $U = \bigcup_{i=1}^{m} [x]_i$, $\forall i \neq j \Rightarrow [x]_i \cap [x]_j = \varnothing$) and $Y = \{[y]_1, [y]_2, \dots, [y]_n\}$.

Quotient sets viz the sets $X/\mathcal{P}$, $X/\mathcal{Q}$ are ipso facto generated either by different algorithms or by the same algorithm with varied parameters. Consider known metrics with the aim of modifying them for use in image segmentation. There exists a type of metrics based on the search for equivalence classes intersections with maxima cardinalities.

These include van Dongen metric (1) and evident transformation of Larsen similarity measure (2)

$$\rho(X,Y) = 2N - \sum_{i=1}^{m} \max_{j\in\{1,2,\dots,n\}} |[x]_i \cap [y]_j| + \sum_{j=1}^{n} \max_{i\in\{1,2,\dots,m\}} |[x]_i \cap [y]_j| \quad (1)$$

$$\rho(X,Y) = \frac{d(X,Y) + d(Y,X)}{2}, \quad (2)$$

It should be emphasized that finding extremes can significantly increase the influence of the corresponding outliers or novelties, thereby distorting the result of the comparison of partitions as a whole.

The Mirkin metric obtained by generalization of the Hamming metric is of undoubted interest

$$\rho(X,Y)=\sum_{i=1}^{m}|[x]_i|^2+\sum_{j=1}^{n}|[y]_j|^2-$$

$$-2\sum_{i=1}^{m}\sum_{j=1}^{n}|[x]_i\cap[y]_j|^2.$$

The Meila metric, based on entropy and known as the information variation, has found wide applications

$$\rho(X,Y)=H(X)+H(Y)-2I(X,Y)$$

where

$$H(X)=-\sum_{i=1}^{m}p([x]_i)\log p([x]_i),$$

$$p([x]_i)=|[x]_i|/N,$$

$$I(X,Y)=\sum_{i=1}^{m}\sum_{j=1}^{n}p([x]_i)p([y]_j)\log\frac{p([x]_i,[y]_j)}{p([x]_i)p([y]_j)},$$

$$p([x]_i,[y]_j)=|[x]_i\cap[y]_j|/N.$$

EMD (originally, transformation cost of one distribution into another one) based comparison of partitions can be used to calculate distances between feature descriptions sets $P=\{(p_1,w_{p_1}),\ldots,(p_m,w_{p_m})\}$ and $Q=\{(q_1,w_{q_1}),\ldots,(q_n,w_{q_n})\}$ that formed on partitions where w_{p_i} and w_{q_j} are the weights. If base similarity measure $d(p_i,q_j)$ is given, then computing of EMD there is nothing else but solution to a linear programming problem

$$\sum_{i=1}^{m}\sum_{j=1}^{n}d(p_i,q_j)c(p_i,q_j)\to\min,$$

$$c(p_i,q_j)\ge 0,\quad 1\le i\le m,\quad 1\le j\le n,$$

$$\sum_{j=1}^{n}c(p_i,q_j)\le w_{p_i},\quad 1\le i\le m,$$

$$\sum_{i=1}^{m}c(p_i,q_j)\le w_{q_j},\quad \sum_{j=1}^{n}c(p_i,q_j)\le w_{p_i},$$

$$\sum_{i=1}^{m}\sum_{j=1}^{n}c(p_i,q_j)=\min\left\{\sum_{i=1}^{m}w_{p_i},\sum_{j=1}^{n}w_{q_j}\right\}$$

where $c(p_i,q_j)$ is a quantity of "product transported" from cluster p_i to cluster q_j considering weights w_{p_i}, $w_{,}q_j$, and $d(p_i,q_j)$ are the transportation costs of the product unit transition. It should be noted, if $\sum_{i=1}^{m}w_{p_i}=\sum_{j=1}^{n}w_{q_j}=1$ and $d(p_i,q_j)$ is a metric, then EMD is a metric also. MiCROM (Minimum-Cost Region Matching) [26] is some development of EMD and models the comparison of segmented images as a minimum-cost network flow problem. The advantage of these metrics is the use of weighting coefficients, which adapts them to image segmentation, while the disadvantage is the high computational complexity with a sufficiently large number of equivalence classes.

Now, let U be an arbitrary measurable set with a measure $\mu(U)<\infty$, i.e., for any $A\subseteq U$ exists some number $\mu(A)$ which is the measure (length, area, volume, mass distribution, probability distribution, cardinality, etc.) then the metric is known [27]

$$\rho(X,Y)=\sum_{k=1}^{m}\sum_{l=1}^{n}(\mu([x]_k\,\Delta\,[y]_l)\mu\cdot$$
$$([x]_k\cap[y]_l)),\qquad(3)$$

where $[x]_k\,\Delta\,[y]_l=([x]_k\backslash[y]_l)\cup([y]_l\backslash[x]_k)$ is a symmetrical difference.

It includes similarity and dissimilarity measures of equivalence classes simultaneously with the simplest computability. Namely, a scalable formalized procedure for the calculation of functional (3) is as follows. Let $S_\Delta(X,Y)=(S_k^{kl})$, $S_\cap(X,Y)=(S_k^{kl})$, where $k=1,m$, $l=1,n$, $S_\Delta^{kl}=\mu([x]_k\,\Delta\,[y]_l)$, $S_\cap^{kl}=\mu([x]_k\cap[y]_l)$. Then the elementwise matrix multiplication $Q_\Delta^{kl}=S_\Delta^{kl}S_\cap^{kl}$ gives $(m\times n)$ matrix $Q_{\Delta\cap}$, the sum of whose elements is $\rho(X,Y)$.

Thus, based on the analysis carried out, metric (3) appears to be the most acceptable for comparing image segmentations, but there exists a need to expand the understanding of the relationship of spatial content with an image. Suppose U is a finite field of view, then it is necessary to prove

$$\rho(X,Y)=\sum_{k=1}^{m}\sum_{l=1}^{n}\varphi([x]_k,[y]_l)\,|[x]_k\Delta[y]_l|$$
$$|[x]_k\cap[y]_l|\qquad(4)$$

where $\varphi([x]_k,[y]_l)$ is a function that characterizes the brightness, color, or texture properties of equivalence classes. The objective of the study is to ground and investigate (4), which we will call a weighted metric. Emphasize that (4) takes into account not only the shape of regions (individual partition elements) and their mutual spatial location, but also the different sensations on the eye in compressed form, i.e. $X=\{(\alpha_1,[x]_1),(\alpha_2,[x]_2),\ldots,(\alpha_m,[x]_m)\}$. Hereafter, we call (4) weighted metric.

The aim of this work is to develop a metric-based loss function for image segmentation and provide its mathematical grounding, which takes into account the geometric and semantic features of regions for training neural networks.

3. METRIC FOR IMAGE SEGMENTATIONS MATCHING

To prove (4) be a metric, it suffices to show that $\rho(X,Y)$ is nonnegative and satisfies the axioms of identity (reflexivity), symmetry, and the triangle inequality.

Denote $\alpha_{ij} = \varphi([x]_i,[y]_j)$. Let $\alpha_{ij} = \alpha_{ji} > 0$, this condition is easy to fulfill at the stage of choosing brightness, color, or texture characteristics. If any $\alpha_{ij} = 0$, then X,Y cease to be partitions, since they do not cover U. Thus (4) is obviously nonnegative and satisfies the symmetry axiom due to the commutativity of symmetric difference and intersection.

To prove reflexivity, it is necessary to show that $\rho(X,Y) = 0 \Leftrightarrow X = Y$. First prove the sufficiency. In accordance with the symmetry property, we have

$$\rho(X,X) = \sum_{k=1}^{m}\sum_{l=1}^{m} \alpha_{kl}\,|[x]_k \triangle [x]_l|\,|[x]_k \cap [x]_l| = \sum_{l=1}^{m} \alpha_{kk}\,|[x]_k \triangle [x]_k|\,|[x]_k \cap [x]_k| + 2\sum_{\substack{k,l=1\\k>l}}^{m} \alpha_{kl}\,|[x]_k \triangle [x]_l|\,|[x]_k \cap [x]_l|$$

Thus, the expression consists of m terms with the same indices, and therefore with the same partition elements, and (m^2-m) terms with different partition elements. The first group of terms obviously consists of zeros, since $|[x]_k \triangle [x]_k| = 0$ for all $k = \overline{1,m}$, and the second group also gives zero terms, because $|[x]_k \cap [x]_l| = 0$ for all $k,l = \overline{1,m}$ provided $k \neq l$.

Now prove the necessity. To do this, assume that for two partitions $X \neq Y$ equality (4) holds. Note, since all terms of functional (4) are nonnegative, it is equal to 0 only if each of them is equal to 0.

Consider several facts that will be useful in proving the triangle inequality. Choose an element $[x]^* \in X$, it is included in the set of "zero" terms $\alpha_{*l} \cdot |[x]^* \triangle [y]_l| \cdot |[x]^* \cap [y]_l|$ where $l = \overline{1,n}$.

Suppose $[x]^*$ does not belong to Y, then for all $[y]_l \in Y$ inequality $|[x]^* \triangle [y]_l| \neq 0$ holds. Then for all indices $l = \overline{1,n}$ the equality $|[x]^* \cap [y]_l| = 0$ must be true. Though it is possible while $[x]^* = \varnothing$ since $[x]^* \in U$ and family $Y = \{[y]_1,[y]_2,\dots,[y]_n\}$ covers the set U. But $[x]^*$ belongs to partition X of the same set U, and of course $[x]^* \neq \varnothing$.

Then there exist elements $[y]_1^*,[y]_2^*,\dots,[y]_p^* \in Y$ which cover subset $[x]^* \subset U$ and have nonempty intersection with it, i.e. $|[x]^* \cap [y]_r^*| \neq 0, r = \overline{1,p}$. Therefore, we get a contradiction – one can assert that any element $[x]^* \in X$ is an element from Y, i.e. $X \subset Y$. By virtue of symmetry, we have $Y \subset X$ and finally $X = Y$, which proves reflexivity.

Consider several facts that will be useful in proving the triangle inequality. For any image U on a power set π_U let us distinguish Π_U be a set of finite (regarding the quantity of equivalence classes) partitions. Each equivalence class $[x]_k \in X \in \Pi_U$ is characterized by some number, i.e. $X = \{(\alpha_1,[x]_1),(\alpha_2,[x]_2),\dots,(\alpha_m,[x]_m)\}$. Obviously, (3) is correct on the whole of Π_U ($\alpha_k \equiv 1, k = \overline{1,m}$), and (4) is a metric on the reduction $P_U \subset \Pi_U$ to be a carrier of metric space, which we have to establish.

For an arbitrary domain $D \subset U$ and any partition $X = \{[x]_1,[x]_2,\dots,[x]_m\}$, since partitioning X splits a set D into disjoint subsets, the cardinality additivity implies the equality

$$|D| = \sum_{i=1}^{m} |D \cap [x]_i|.$$

For the reason that we consider the elements of the partition with some positive weights, a required metric space has to be constructed on the following carrier: $\mathbb{R}_m^+ \times \Pi_U$. Then it is easily seen that for any $D \subset U$ and $X \in \mathbb{R}_m^+ \times \Pi_U$ it follows that

$$|D| = \sum_{i=1}^{m} \alpha_i\,|D \cap [x]_i|\,. \qquad (5)$$

In fact, equality (5) is a hyperplane in $\mathbb{R}_m$. Denote by $K_m^+(X)$ its intersection with $\mathbb{R}_m^+$, since it essentially depends on the partition $X = \{(\alpha_1,[x]_1),(\alpha_2,[x]_2),\dots,(\alpha_m,[x]_m)\}$.

The important point to note here is the possibility to rewrite (4) in tantamount form

$$\rho(X,Y) = \sum_{k=1}^{m} |[x]_k|^2 + \sum_{l=1}^{n} |[y]_l|^2 - 2\sum_{k=1}^{m}\sum_{l=1}^{n} \alpha_{kl}\,|[x]_k \cap [y]_l|^2 \,. \qquad (6)$$

One may conjecture that for any sets $A,B \in U$ there exists a relationship

$$|A \triangle B| = |A| + |B| - 2|A \cap B|. \quad (7)$$

Indeed, taking into account obvious equalities $A = (A/B) \cup (A \cap B)$ and $B = (B/A) \cup (A \cap B)$, what is more, sets A/B and $A \cap B$ do not intersect, from cardinality additivity we get $|A| = |A/B| + |A \cap B|$ and $|B| = |B/A| + |A \cap B|$. Summing up these equalities and considering $A \triangle B = (A/B) \cup (B/A)$, we get $|A| + |B| = |A/B| + |B/A| + 2|A \cap B| = |A \triangle B|$ which is equivalent to (7).

Further applying (7) to equivalence classes $[x]_k$ and $[y]_l$, we represent (4) in the form

$$\rho(X,Y) = \sum_{k=1}^{m}\sum_{l=1}^{n}\alpha_{kl}(|[x]_k| + |[y]_l| - 2|[x]_k \cap [y]_l|) \cdot \cdot |[x]_k \cap [y]_l| =$$
$$= \sum_{k=1}^{m}\sum_{l=1}^{n}\alpha_{kl}|[x]_k| \cdot |[x]_k \cap [y]_l| +$$
$$+ \sum_{k=1}^{m}\sum_{l=1}^{n}\alpha_{kl}|[y]_l| \cdot |[x]_k \cap [y]_l| -$$
$$-2\sum_{k=1}^{m}\sum_{l=1}^{n}\alpha_{kl}|[x]_k \cap [y]_l|^2.$$

Now select in the last equality the sums of the form $\sum_{l=1}^{n}\alpha_{kl}|[x]_k \cap [y]_l|$, $\sum_{k=1}^{m}\alpha_{kl}|[x]_k \cap [y]_l|$, then accordingly (5) under restrictions $H_U(X,Y) = \{K_m^+(X) \cap K_n^+(Y)\} \times P_U \subset \pi_U$ hold $\sum_{l=1}^{n}\alpha_{kl}|[x]_k \cap [y]_l| = |[x]_k|$ and $\sum_{k=1}^{m}\alpha_{kl}|[x]_k \cap [y]_l| = |[y]_l|$ for $k = \overline{1,m}, l = \overline{1,n}$ what means the fulfillment of equality (6) and establishes identicalness of (4) and (6).

Since we will have to consider various intersections of $X, Y \in P_U$ below, we will point out that all $X \cap Y = \{[x]_k \cap [y]_l\}_{k=\overline{1,m},l=\overline{1,n}}$ are also partitions. In fact, if we select an arbitrary element $u \in U$ then since X, Y are partitions, there will be indices $k \in \{1,2,\dots,m\}$, $l \in \{1,2,\dots,n\}$ for which $u \in [x]_k$, $u \in [y]_l$, i.e. $u \in [x]_k \cap [y]_l$. Consequently, $\cup_{k=1}^{m}\cup_{l=1}^{n}\{[x]_k \cap [y]_l\} = U$.

On the other hand, due to the associativity and commutativity of the intersection operation of sets, we can specify

$$([x]_k \cap [y]_l) \cap ([x]_{k'} \cap [y]_{l'}) = ([x]_k \cap [x]_{k'}) \cap ([y]_l \cap [y]_{l'}).$$

But since the pairs (k,l) and (k',l') are not equal, then either $k \neq l$, or $k' \neq l'$, or these inequalities are satisfied simultaneously, which means that one of the sets $[x]_k \cap [y]_l$ or $[x]_{k'} \cap [y]_{l'}$ is the empty one, since they belong to the partitions X and Y respectively. Thus, for any pairs $(k,l) \in \{1,2,\dots,m\} \times \{1,2,\dots,n\}$ we end up with $([x]_k \cap [y]_l) \cap ([x]_{k'} \cap [y]_{l'}) = \varnothing$.

As a result, we conclude that a family of sets $X \cap Y = \{[x]_k \cap [y]_l\}_{k=\overline{1,m},l=\overline{1,n}}$ is an exhaust set of non-empty subsets such that every element is in exactly one of these subsets, i.e. $X \cap Y \in P_U$.

Just now consider intersections of three arbitrary partitions $X, Y, Z \in P_U$, where

$$\begin{cases} X = \{(\alpha_1,[x]_1),(\alpha_2,[x]_2),\dots,(\alpha_m,[x]_m)\}, \\ Y = \{(\beta_1,[y]_1),(\beta_2,[y]_2),\dots,(\beta_n,[y]_n)\}, \\ Z = \{(\gamma_1,[z]_1),(\gamma_2,[z]_2),\dots,(\gamma_p,[z]_p)\}. \end{cases}$$

Withal, for each pair of sets of (generally speaking normalized) weights, there exists a conformity

$$\begin{cases} \{\alpha_1,\alpha_2,\dots,\alpha_m\},\{\beta_1,\beta_2,\dots,\beta_n\} \Rightarrow \lambda_{kl} = \varphi([x]_k,[y]_l), \\ \{\alpha_1,\alpha_2,\dots,\alpha_m\},\{\gamma_1,\gamma_2,\dots,\gamma_p\} \Rightarrow \mu_{ki} = \varphi([x]_k,[z]_i), \\ \{\beta_1,\beta_2,\dots,\beta_n\},\{\gamma_1,\gamma_2,\dots,\gamma_p\} \Rightarrow \theta_{li} = \varphi([y]_l,[z]_i). \end{cases}$$

Denote $d_{kli} = |[x]_k \cap [y]_l \cap [z]_i|$, $k = \overline{1,m}$, $l = \overline{1,n}, i = \overline{1,p}$. In accordance with the property above, consider the partition $X \cap Y$ and apply (5), assuming that D is a set $[z]_i$ while skipping weights, then

$$|[z]_i| = \sum_{k=1}^{m}\sum_{l=1}^{n}|([z]_i \cap ([x]_k \cap [y]_l))| = \sum_{k=1}^{m}\sum_{l=1}^{n}d_{kli}. \quad (8)$$

Reasoning in a completely similar way and using as D in (5) in turn the sets $[x]_k \cap [y]_l$, $[x]_k \cap [z]_i$, $[y]_l \cap [z]_i$, and as the corresponding partition X, Y, Z, we obtain the validity of expressions

$$\begin{cases} |[x]_k \cap [y]_l| = \sum_{i=1}^{p} d_{kli}, \\ |[x]_k \cap [z]_i| = \sum_{l=1}^{m} d_{kli}, \\ |[y]_l \cap [z]_i| = \sum_{k=1}^{m} d_{kli}. \end{cases} \quad (9)$$

If triplet $X, Y, Z \in P_U$ is such that $X, Y \subset H_U(X,Y)$, $X, Z \subset H_U(X,Z)$, $Y, Z \subset H_U(Y,Z)$, then from (6) it follows immediately

$$\rho(X,Y) = \sum_{k=1}^{m}|[x]_k|^2 + \sum_{l=1}^{n}|[y]_l|^2 - 2\sum_{k=1}^{m}\sum_{l=1}^{n}\lambda_{kl}|[x]_k \cap [y]_l|^2,$$

$$\rho(X,Z)=\sum_{k=1}^{m}|[x]_k|^2+\sum_{i=1}^{p}|[z]_i|^2-$$
$$-2\sum_{k=1}^{m}\sum_{i=1}^{p}\mu_{ki}\,|[x]_k\cap[z]_i|^2,$$
$$\rho(Y,Z)=\sum_{l=1}^{n}|[y]_l|^2+\sum_{i=1}^{p}|[z]_i|^2-$$
$$-2\sum_{l=1}^{n}\sum_{i=1}^{p}\theta_{li}\,|[y]_l\cap[z]_i|^2.$$

The preparatory reasoning carried out allows us to prove the triangle inequality. From the relationships above, we get

$$\frac{1}{2}\{\rho(X,Z)+\rho(Y,Z)-\rho(X,Y)\}=$$
$$=\sum_{i=1}^{p}|[z]_i|^2-\sum_{k=1}^{m}\sum_{i=1}^{p}\mu_{ki}\,|[x]_k\cap[z]_i|^2-$$
$$-\sum_{l=1}^{n}\sum_{i=1}^{p}\theta_{li}\,|[y]_l\cap[z]_i|^2+$$
$$+\sum_{k=1}^{m}\sum_{l=1}^{n}\lambda_{kl}\,|[x]_k\cap[y]_l|^2.$$

Taking into account (8) and (9), we obtain that the triangle inequality is fulfilled if and only if

$$\sum_{i=1}^{p}\left(\sum_{k=1}^{m}\sum_{l=1}^{n}d_{kli}\right)^2+$$
$$+\sum_{k=1}^{m}\sum_{l=1}^{n}\lambda_{kl}\left(\sum_{i=1}^{p}d_{kli}\right)^2\geq$$
$$\geq\sum_{k=1}^{m}\sum_{i=1}^{p}\mu_{ki}\left(\sum_{l=1}^{n}d_{kli}\right)^2+ \qquad (10)$$
$$+\sum_{l=1}^{n}\sum_{i=1}^{p}\theta_{li}\left(\sum_{k=1}^{m}d_{kli}\right)^2.$$

Let us fix $p=1$ and denote $g_{kl}=d_{k|1}$, noting that $g_{kl}\geq 0$, then the last inequality implies

$$\left(\sum_{k=1}^{m}\sum_{l=1}^{n}g_{kl}\right)^2+\sum_{k=1}^{m}\sum_{l=1}^{n}\lambda_{kl}\,(g_{kl})^2\geq$$
$$\geq\sum_{k=1}^{m}\mu_{k1}\left(\sum_{l=1}^{n}g_{kl}\right)^2 \qquad (11)$$
$$+\sum_{l=1}^{n}\theta_{l1}\left(\sum_{k=1}^{m}g_{kl}\right)^2.$$

Take cognizance of the set of numbers g_{kl} as an $(m\times n)$ matrix G explanation, then the right-hand side of this inequality in brackets contains the total of the row elements and the total of the column elements, respectively.

The structure of the left-hand side of the inequality has the form

$$2\sum_{k=1}^{m}\sum_{l=1}^{n}(1+\lambda_{kl})(g_{kl})^2+\sum_{(k,l)\neq(k',l')}g_{kl}\,g_{k'l'}$$

and the right side looks like this

$$2\sum_{k=1}^{m}\sum_{l=1}^{n}(\mu_{k1}+\theta_{l1})(g_{kl})^2+\sum_{k=1}^{m}\mu_{k1}\sum_{l\neq l`}g_{kl}g_{kl`}+$$
$$+\sum_{l=1}^{n}\theta_{l1}\sum_{k\neq k`}g_{kl}g_{k`l}.$$

Comparing them, emphasize that they have a common part (double sum of squares), the left part contains all pairwise products of matrix G elements with unequal indices, and the right part contains all pairwise products with unequal indices, but from one row or from one column (i.e. there are no pairwise products, specifically diagonal elements). Thus, the number of terms on the right side is less than on the left side, and all these terms are contained in the left side, i.e. inequality (10) is satisfied for unit weights. In general (10) is true if

$$\begin{cases}\lambda_{kl}+1\geq\mu_{k1}+\theta_{l1},\\ \sum\limits_{k=1}^{m}\sum\limits_{l\neq l'}g_{kl}g_{kl'}\geq\sum\limits_{k=1}^{m}\mu_{k1}\sum\limits_{l\neq l'}g_{kl}\,g_{kl'},\\ \sum\limits_{l=1}^{n}\sum\limits_{k\neq k'}g_{kl}g_{k'l}\geq\sum\limits_{l=1}^{n}\theta_{l1}\sum\limits_{k\neq k'}g_{kl}\,g_{k'l}.\end{cases}$$

What does this system of inequalities represent? It is easily seen that there are $2m+2n$ variables in the space R^{2m+2n}, which is divided into two parts by hyperplanes of the form

$$\begin{cases}\lambda_{kl}-\mu_{k1}+\theta_{l1}+1=0,\\ \sum\limits_{k=1}^{m}(1-\mu_{k1})\sum\limits_{l\neq l'}g_{kl}g_{kl'}=0,\\ \sum\limits_{l=1}^{n}(1-\theta_{l1})\sum\limits_{k\neq k'}g_{kl}g_{k'l}=0.\end{cases}$$

The intersection of these parts and the first quadrant of R^{2m+2n} gives the desired region $H_1(X,Y,Z)$, which, of course, depends on the

original triple of partitions. Thus, inequality (11) takes place in the region $\Phi_1 = H_1(X,Y,Z)\times(X,Y,Z)\subset P_U$ at M. The regions $\Phi_2,\Phi_3,...,\Phi_p$ are constructed similarly. There is no difficulty in understanding that (10) transforms into the form

$$\sum_i\left(\sum_{k,l} g_{ki}^l\right)^2+\sum_{k,l}\lambda_{kl}\left(\sum_i g_{ki}^l\right)^2\geq$$

$$\geq\sum_{k,i}\mu_{ki}\left(\sum_j g_{ki}^j\right)^2+\sum_{l,i}\theta_{li}\left(\sum_k g_{ki}^l\right)^2,$$

where the numbers d_{kli} from (10) are considered as a set of p matrices $G_i=(g_{ki}^l)_{k,l=1}^{m,n}$, $i=\overline{1,p}$.

When adding over i in accordance with the scheme, we obtain at every step the required hyperplanes and finally $\Phi=\bigcap_{i=1}^p\Phi_i$. Thus, we conclude that in the region $\mathrm{P_U(X,Y,Z)}=\Phi\times(\mathrm{X,Y,Z})\subset\mathrm{P_U}$ the triangle inequality is satisfied, i.e. weighted functional (2) is a metric for image segmentations matching.

To this end, the results obtained need careful explanations. The metric space on the set of weight partitions $\mathrm{P_U}$ is formed on a certain restriction $\mathrm{P'_U}\subset\mathrm{P_U}$, consisting of two parts. In the first part, the metric does not depend on the weight coefficients, and this subdomain $\mathrm{P_U^1(X,Y)}\subset\mathrm{P'_U}$ is the union of subdomains by any pairs of weighted partitions (X,Y). In the second part, the metric depends on the weight coefficients, and this subdomain $\mathrm{P_U^2(X,Y,Z)}\subset\mathrm{P'_U}$ is the union of subdomains by any triples of weighted partitions (X,Y,Z), and finally $\mathrm{P'_U}=\mathrm{P_U^1}\bigcup\mathrm{P_U^2}$. Of course, we would like to find a metric on the entire space $\mathrm{P_U}$, and not on the restriction $\mathrm{P'_U}\subset\mathrm{P_U}$, but, in our opinion, it seems impossible to tackle the problem successfully although further experiments may yield intriguing results.

4. EXPERIMENTS

This section will show how it is possible to apply our metric to the segmentation. We have taken the SegNet architecture, which is supposed to solve a pixel-by-pixel regression. As a dataset, we created a synthetic set of five shapes: circle, rectangle, triangle, pentagon, star; that do not overlap in a 100 × 100px image. The number of shapes in the dataset is normally distributed, and there can be from two to five pieces in each image. These parameters are also customizable. In order not to fix the number of classes, we decided that the model would output a single-channel image. To satisfy the metric's requirement for strict partitioning – where each pixel belongs to exactly one class, including the background, while accounting for object connectivity – we designed the architecture accordingly and created a synthetic dataset.

The first problem to be solved is to select elements (shapes) for partitioning. We need to consider the connectivity of these elements and the fact that the output will not be discrete. Furthermore, the complexity of the task is compounded by the fact that the selection operation must be differentiable. If the operation is not differentiable or there is no differentiable approximation, we will lose gradients and will not be able to train the network. For selection, we decided to use KNN with element connectivity checking since the number of elements is not fixed. At the beginning of training, the number of clusters can be quite large (Fig. 1), which in turn requires sufficient computational resources.

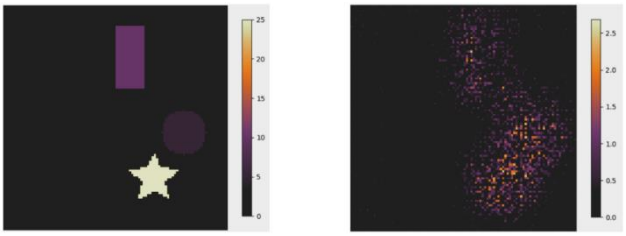

Fig. 1. **Input and output of the first processing step**
Source: **compiled by the authors**

Since the KNN algorithm itself is not differentiable (relies on discrete, non-differentiable operations such as hard distance ranking and argmin assignments), we used the STE trick [28], [29], [30], which allowed us to pass gradients backward. The STE resolves this by retaining the strict, hard class assignments during the forward pass, which is essential for the metric's partitioning requirements, while acting as an identity function during the backward pass to route gradients directly through the non-differentiable steps. The choice of such a feature also determines the learning result. We, as an experiment, used 1 × 1 convolution, Gumbel-Softmax, and soft masks. Soft masks describe the probabilistic value of whether a pixel belongs to hard centroids. The intermediate training results are shown in Fig. 2.

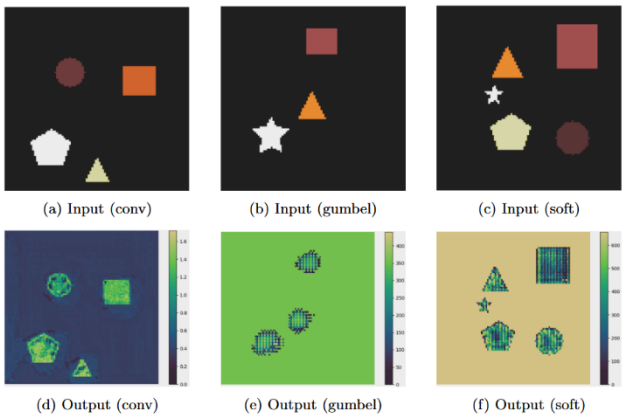

Fig. 2. **Comparison of different approximation methods: conv 1×1, Gumbel-Softmax, and soft masks**
Source: **compiled by the authors**

As can be seen, the function does not train the network to segment objects by class, but allows it to find objects and emphasize the separation of areas, while taking connectivity into account. This can be useful in combined loss functions. Fig. 3 illustrate the KNN outputs and the differences in the results obtained during training for various STE.

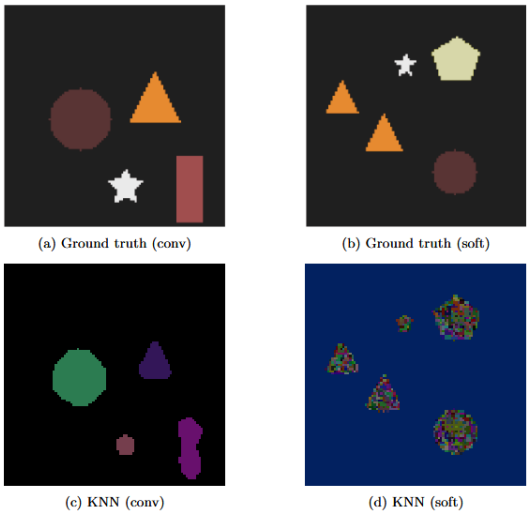

Fig. 3. **Examples of different KNN outputs with different STE**
Source: **compiled by the authors**

Additionally, before starting the training process, we checked the gradients (Fig. 4) the input image was passed through the KNN module, and its output was then directed to the loss function for the backward pass. It should be noted that, due to computing the intersection between the predicted result and the ground truth in the loss function, we had to zero out the gradients when the metric equals zero. This was necessary because the self-intersection would produce non-zero gradients, which would negatively impact the training process.

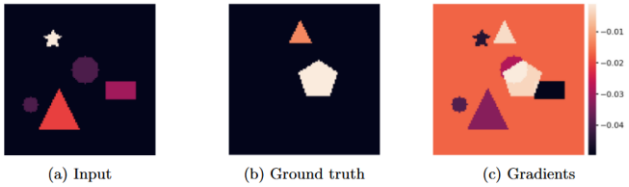

Fig. 4. **Visualization of computed gradients during the backward pass**
Source: **compiled by the authors**

5. DISCUSSIONS

We proposed a weighted metric for comparing image partitions that accounts for both the geometry of regions and their semantic properties (brightness, color, texture) through a weight function. Compared to widely used functions (Dice/Tversky, IoU, Lovász-Softmax, etc.) and boundary-based ones (Boundary/Hausdorff), ours fundamentally differs in that it defines precisely a metric on the space of partitions, whereas the mentioned functions either are not metrics or require special smoothing and do not satisfy the triangle inequality. This property enables hierarchical search over the space of partitions and more informed hyperparameter tuning. Unlike EMD and MiCROM, where accuracy is achieved through solving linear programming problems, the proposed metric reduces to matrix computations and is therefore potentially simpler computationally with a moderate number of regions. However, at the initial stages of training, the number of regions is typically large, which increases memory requirements. The practical part revealed a key bottleneck. To use the metric as an end-to-end loss function, it is necessary to differentiably extract non-overlapping segments. We applied KNN clustering with connectivity checking and the Straight-Through Estimator to approximate non-differentiable steps. On a synthetic dataset of non-overlapping shapes, the model optimized with our metric was able to separate objects accounting for connectivity, as shown in intermediate results in Fig. 2. This makes the metric a good candidate for combined loss functions that will emphasize correct partition topology, to which functions focusing on semantics or class membership can be added. It should also be noted that the choice and normalization of directly affect sensitivity to object sizes and contrast, as different tasks may require different outcomes.

6. CONCLUSIONS

This work presents a weighted metric for comparing segmentations that operates with partitions and takes into account semantic and visual

features. We adapted it for use as a loss function using the STE trick. Existing loss functions were compared, and their strengths, weaknesses, and areas of applicability were identified. For the proposed metric, we mathematically proved the metric axioms, particularly the triangle inequality, and conducted experiments on a synthetic dataset confirming its feasibility. The proposed metric bridges partition theory with the practice of training segmentation models. Initial results confirm the applicability of the approach, albeit with limitations. Future work requires investigations aimed at improving differentiable approximations for object extraction, either by replacing them with learnable object detection methods. Furthermore, evaluation and comparative analysis on real-world tasks against existing loss functions should be conducted.

7. ACKNOWLEDGEMENTS

The results of this research were obtained under the international research project INITIATE under the grant No.101136775-HORIZON-WIDERA-2023-ACCESS-03.

REFERENCES

1. Long, J., Shelhamer, E. & Darrell, T. “Fully convolutional networks for semantic segmentation”. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015. p. 3431–3440, https://www.scopus.com/pages/publications/84959205572. DOI: https://doi.org/10.1109/CVPR.2015.7298965.

2. Ronneberger, O., Fischer, P. & Brox, T. “U-Net: Convolutional networks for biomedical image segmentation”. *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, Lecture Notes in Computer Science*. 2015; 9351: 234–241, https://www.scopus.com/pages/publications/84951834022. DOI: https://doi.org/10.1007/978-3-319-24574-4_28.

3. Badrinarayanan, V., Kendall, A. & Cipolla, R. “SegNet: A deep convolutional encoder–decoder architecture for image segmentation”. *IEEE Transactions on Pattern Analysis and Machine Intelligence.* 2017; 39 (12): 2481–2495, https://www.scopus.com/pages/publications/85033697420. DOI: https://doi.org/10.1109/tpami.2016.2644615.

4. He, K., Gkioxari, G., Dollar, P. & Girshick, R. “Mask R-CNN”. *Proceedings of the IEEE International Conference on Computer Vision (ICCV).* 2017. p. 2980–2988, https://www.scopus.com/pages/publications/85040313738. DOI: https://doi.org/10.1109/iccv.2017.322.

5. Ren, S., He, K., Girshick, R. & Sun, J. “Faster R-CNN: Towards real-time object detection with region proposal networks”. *Advances in Neural Information Processing Systems (NeurIPS)*. 2015; 28: 91–99, https://www.scopus.com/pages/publications/84960980241.

6. Takikawa, T., Acuna, D., Jampani, V. & Fidler, S. “Gated-SCNN: Gated Shape CNNs for Semantic Segmentation”. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019. p. 5228–5237, https://www.scopus.com/pages/publications/85081936527.
DOI: https://doi.org/10.1109/iccv.2019.00533.

7. Wu, H., Zhang, J., Huang, K., Liang, K. & Yu, Y. “FastFCN: Rethinking dilated convolution in the backbone for semantic segmentation”. *arXiv*. 2019. DOI: https://doi.org/10.48550/arXiv.1903.11816.

8. Cheng, B., Schwing, A. G. & Kirillov, A. “Per-Pixel classification is not all you need for semantic segmentation”. *Advances in Neural Information Processing Systems (NeurIPS)*. 2021; 34: 17864–17875, https://www.scopus.com/pages/publications/85132049738. DOI: https://doi.org/10.48550/arXiv.2107.06278.

9. Cheng, B., Misra, I., Schwing, A. G., Kirillov, A. & Girdhar, R. “Masked-Attention Mask Transformer for Universal Image Segmentation”. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022. p. 1280–1289, https://www.scopus.com/pages/publications/85141814465. DOI: https://doi.org/10.1109/cvpr52688.2022.00135.

10. Jadon, S. “A survey of loss functions for semantic segmentation”. *IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*. 2020. p. 1–7, https://www.scopus.com/pages/publications/85099054674.
DOI: https://doi.org/10.1109/CIBCB48159.2020.9277638.

11. Mohseni Salehi, S. S., Erdogmus, D. & Gholipour, A. “Tversky loss function for image segmentation using 3D fully convolutional deep networks”. *International Workshop on Machine Learning in Medical Imaging (MLMI), in Conjunction with MICCAI*. 2017, https://www.scopus.com/pages/publications/85029684537. DOI: https://doi.org/10.1007/978-3-319-67389-9_44.

12. Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I. & Savarese, S. “Generalized intersection over union: A metric and a loss for bounding box regression”. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019. p. 658–666, https://www.scopus.com/pages/publications/85074449910. DOI: https://doi.org/10.1109/CVPR.2019.00075.

13. Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R. & Ren, D. “Distance-IoU loss: faster and better learning for bounding box regression”. *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. 2020. p. 12993–13000, https://www.scopus.com/pages/publications/85106618747.
DOI: https://doi.org/10.1609/aaai.v34i07.6999.

14. Zheng, Z., Wang, P., Ren, D., Liu, W., Ye, R., Hu, Q. & Zuo, W. “Enhancing geometric factors in model learning and inference for object detection and instance segmentation”. *IEEE Transactions on Cybernetics*. 2022; 52 (8): 3349–3361, https://www.scopus.com/pages/publications/85113909734. DOI: https://doi.org/10.48550/arXiv.2005.03572.

15. Wang, Z., Ning, X. & Blaschko, M. B. “Jaccard Metric Losses: Optimizing the Jaccard Index with Soft Labels”. *Advances in Neural Information Processing Systems*. 2023; 36, https://www.scopus.com/pages/publications/85191158441. DOI: https://doi.org/10.48550/arXiv.2302.05666.

16. Berman, M., Triki, A. R. & Blaschko, M. B. “The Lovász-Softmax loss: A tractable surrogate for the optimization of the Intersection-over-Union measure in neural networks”. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018. p. 4413–4421, https://www.scopus.com/pages/publications/85062843018. DOI: https://doi.org/10.1109/CVPR.2018.00464.

17. Kervadec, H., Dolz, J., Tang, M., Granger, E., Boykov, Y. & Bloch, I. “Boundary loss for highly unbalanced segmentation”. *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. 2019; 11766: 285–293, https://www.scopus.com/pages/publications/85092711864. DOI: https://doi.org/10.1016/j.media.2020.101851.

18. Karimi, D. & Salcudean, S. E. “Reducing the Hausdorff distance in medical image segmentation with convolutional neural networks”. *IEEE Transactions on Medical Imaging*. 2020; 39 (2): 499–513, https://www.scopus.com/pages/publications/85079020336. DOI: https://doi.org/10.1109/TMI.2019.2930068.

19. Schroff, F., Kalenichenko, D. & Philbin, J. “FaceNet: A unified embedding for face recognition and clustering”. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015. p. 815–823, https://www.scopus.com/pages/publications/84946751287. DOI: https://doi.org/10.1109/CVPR.2015.7298682.

20. Wen, Y., Zhang, K., Li, Z. & Qiao, Y. “A Discriminative feature learning approach for deep face recognition”. *European Conference on Computer Vision (ECCV)*. 2016. p. 499–515, https://www.scopus.com/pages/publications/84990032583. DOI: https://doi.org/10.1007/978-3-319-46478-7_31.

21. van Dongen, S. “Performance criteria for graph clustering and Markov cluster experiments”. *Amsterdam: Stichting Mathematisch Centrum*. 2000.

22. Jiang, X., Marti, C., Irniger, C. & Bunke, H. “Distance measures for image segmentation evaluation”. *EURASIP Journal on Applied Signal Processing*. 2006; 2006 (1): 035909, https://www.scopus.com/pages/publications/33645661245. DOI: https://doi.org/10.1155/ASP/2006/35909.

23. Mirkin, B. “Mathematical classification and clustering”. *New York: Kluwer Academic Publishers*. 1996.

24. Meilă, M. “Comparing clusterings by the variation of information”. In *B. Schölkopf & M. K. Warmuth (Eds), Computational Learning Theory and Kernel Machines. Berlin, Heidelberg: Springer*. 2003.p. 173–187, https://www.scopus.com/pages/publications/9444274777. DOI: https://doi.org/10.1007/978-3-540-45167-9_14.

25. Rubner, Y., Tomasi, C. & Guibas, L. J. “The Earth Mover’s Distance as a metric for image retrieval”. *International Journal of Computer Vision*. 2000; 40 (2): 99–121, https://www.scopus.com/pages/publications/0034313871. DOI: https://doi.org/10.1023/A:1026543900054.

26. Stehling, R. O., Nascimento, M. A. & Falcão, A. X. “MiCRoM: A metric distance to compare segmented images”. In *S.-K. Chang, Z. Chen, & S.-Y. Lee (Eds), VISUAL 2002. Berlin, Heidelberg: Springer*. 2002. p. 12–23, https://www.scopus.com/pages/publications/84944386853. DOI: https://doi.org/10.1007/3-540-45925-1_2.

27. Kinoshenko, D., Mashtalir, V. & Shlyakhov, V. “A partition metric for clustering features analysis”. *Information Theories and Applications*. 2007; 14 (3): 230–236.

28. Bengio, Y., Léonard, N. & Courville, A. “Estimating or propagating gradients through stochastic neurons for conditional computation”. *arXiv*. 2013. DOI: https://doi.org/10.48550/arXiv.1308.3432.

29. Jang, E., Gu, S. & Poole, B. “Categorical reparameterization with Gumbel-Softmax”. *5th International Conference on Learning Representations (ICLR)*. 2017, https://www.scopus.com/pages/publications/85088225686. DOI: https://doi.org/10.48550/arXiv.1611.01144.

30. Yin, P., Lyu, J., Zhang, S., Osher, S., Qi, Y. & Xin, J. “Understanding straight-through estimator in training activation quantized neural nets”. *arXiv*. 2019, https://www.scopus.com/pages/publications/85083949976. DOI: https://doi.org/10.48550/arXiv.1903.05662.

Conflicts of Interest: The authors declare that they have no conflict of interest regarding this study, including financial, personal, authorship, or other interests, which could influence the research and its results presented in this article.

Received 12.01.2026
Received after revision 13.03.2026
Accepted 19.03.2026

DOI: https://doi.org/10.15276/aait.09.2026.12
УДК 004:83

Метричні функції втрат для сегментації зображень на основі згорткових нейронних мереж

Ковтуненко Андрій Романович[1)]
ORCID: https://orcid.org/0009-0004-9072-7779; andrii.kovtunenko@nure.ua. Scopus Author ID: 58362751200
Машталір Сергій Володимирович[1)]
ORCID: https://orcid.org/0000-0002-0917-6622; sergii.mashtalir@nure.ua. Scopus Author ID: 36183980100
[1)] Харківський Національний Університет Радіоелектроніки, пр. Науки, 14. Харків, 61166, Україна

АНОТАЦІЯ

Сегментація зображень залишається фундаментальною проблемою комп'ютерного зору, при цьому навчання нейронних мереж значною мірою залежить від відповідних функцій втрат. Хоча такі поширені функції втрат, як коефіцієнт Дайса (Dice), широко використовуються, але їм бракує математичного обґрунтування властивостей метрик, і вони не повною мірою враховують геометричну структуру розбиттів (partitions). **Ми пропонуємо зважену метрику** для порівняння сегментацій на основі теорії розбиттів, яка задовольняє всі аксіоми метрики, включаючи нерівність трикутника. Запропонована метрика порівнює розбиття за допомогою операцій симетричної різниці та перетину, враховуючи як просторову структуру, так і семантичні ознаки за допомогою вагової функції, що характеризує властивості регіонів, такі як колір, текстура та інші. **Доводимо, що запропонований** функціонал утворює належний метричний простір на зважених розбиттях за визначених умов, з особливим акцентом на доведенні нерівності трикутника. **Експериментальна перевірка** на синтетичних задачах демонструє доцільність підходу, хоча його практична реалізація стикається з певними викликами, такими як необхідність диференційованого виділення сегментів. Цю проблему можна вирішити за допомогою STE (Straight-Through Estimator). Властивість нерівності трикутника відкриває можливості для застосування ієрархічних підходів до сегментації та ефективного пошуку розбиттів. Ця робота усуває розрив між геометричною теорією кластеризації та сегментацією на основі глибокого навчання, надаючи теоретично обґрунтовану альтернативу евристичним функціям втрат, а також експериментально доводить можливість використання запропонованої метрики в якості функції втрат при навчання згорткових нейронних мереж.

Ключові слова: сегментація зображень; функції втрат; згорткові нейронні мережі; глибоке навчання; метрика; розбиття; комп'ютерний зір

ABOUT THE AUTHORS

Andrii R. Kovtunenko - PhD student of Informatics Department, Kharkiv National University of Radio Electronics.14, Nauky Ave. Kharkiv, 61166, Ukraine
ORCID: https://0009-0004-9072-7779; andrii.kovtunenko@nure.ua. Scopus Author ID: 58362751200
***Research field*:** Image and video processing; data analysis

Ковтуненко Андрій Романович - аспірант кафедри Інформатики Харківського національного університету радіоелектроніки, проспект Науки 14. Харків, 61166, Україна

Sergii V. Mashtalir - Doctor of Engineering Science, Professor, Informatics Department. Kharkiv National University of Radio Electronics.14, Nauky Ave. Kharkiv, 61166, Ukraine
ORCID: https://orcid.org/0000-0002-0917-6622; sergii.mashtalir@nure.ua. Scopus Author ID: 36183980100
***Research field*:** Image and video processing; data analysis

Машталір Сергій Володимирович - доктор технічних наук, професор кафедри Інформатики Харківського національного університету радіоелектроніки, пр. Науки 14. Харків, 61166, Україна