Networks and Heterogeneous Media

https://www.aimspress.com/journal/nhm

NHM, 21(3): 1069–1098.
DOI: 10.3934/nhm.2026044
Received: 25 December 2025
Revised: 13 April 2026
Accepted: 08 May 2026
Published: 21 May 2026

Research article

Choquet integral based on copulas and its application in neural networks

Hong Yang* **and Jiaxue Wei**

College of Mathematics and Statistics, Northwest Normal University, Lanzhou 730070, PR China

* **Correspondence:** Email: ysin888@163.com; Tel: +8613619316908.

Abstract: Usually, convolutional neural networks (CNNs) reduce the features extracted from convolutional layers by adopting the maximum value or arithmetic average method, which is called the pooling process. However, these two pooling methods overlook the spatial dependencies between image features. The Choquet integral is a nonlinear integral, which addresses this limitation by incorporating weighted aggregation via a fuzzy measure, enabling it to effectively model interactions between variables. This capability is essential for processing data with complex dependency structures. Therefore, this paper proposes a novel Choquet integral pooling method for feature extraction in the pooling layer. By conducting experiments on datasets, we demonstrated the effectiveness of the proposed method and compared it with existing techniques. The experimental results indicated that for cases where image features are less prominent and exhibit spatial dependencies, the Choquet integral pooling outperforms traditional pooling methods and demonstrates greater robustness. To ensure the rationality and validity of applying the Choquet integral in CNNs, we conducted a more in-depth study of the Choquet integral based on a copula, that is, the C_C-integral, including its averaging, idempotence, translation invariance, and positive homogeneity. This research not only enriches the application fields of the Choquet integral but also provides new theoretical support and technical paths for the research of neural networks.

Keywords: fuzzy analysis; Choquet integral; C_C-integral; convolutional neural networks

1. Introduction

Aggregation functions formalize the aggregation process [1], providing a function definition that fuses the information represented by multiple values into a single value within the interval [0,1], with two important properties ensuring consistency between the information output and the information carried by the input. Aggregation functions have been applied in many fields, such as multiple attribute and group decision-making [1], classification [2], image processing [3], fuzzy systems and control [4], hierarchical information fusion [5], and the pooling layer of deep learning [6].

Among various aggregation functions, the Choquet integral is a particularly important nonlinear integral that generalizes the classical Lebesgue integral. It was introduced by French mathematician Choquet in 1954 based on capacity theory [7], where a capacity is a set function satisfying monotonicity and continuity. The Choquet integral is an averaging aggregation function, meaning its output lies between the minimum and maximum values of the inputs. It plays a crucial role in applications where interactions between the elements to be aggregated need to be considered [8–10]. Since its introduction, generalized forms of the Choquet integral have attracted increasing attention. In 1987, Sugeno and Murofushi proposed a generalized form of Choquet integrals in [11]. Subsequently, in 1995, Mesiar and Grabisch provided other significant methods and discussed various boundary conditions in [12, 13]. In 2013, Barrenechea et al. applied the Choquet integral as an aggregation function in classification systems based on fuzzy rules [2]. In 2016, Lucca et al. researched the new Choquet integral extension of the aggregation functions in [14]. These extensions of the Choquet integral were named, for example, the C_C-integral [15], C_F-integral [16], and C_T-integral [17]. For the C_C-integral, the methodology used in these generalizations is simply to replace the product in the Choquet integral with a copula, resulting in a pre-aggregation function. The C_C-integral satisfies all the conditions required for an aggregation function.

Convolutional neural networks (CNNs) are an important branch in the field of deep learning. Since their proposal in the 1990s, they have achieved tremendous development and success. With their increasingly widespread application in various fields, they have become a research hotspot in the field of artificial intelligence. Currently, CNNs have been widely applied in image classification [18], object detection [19], and natural language processing [20], and achieve significant results. In image classification, they can automatically learn feature representations of images, achieving high-precision image classification. In object detection tasks, combined with proposal networks, they can achieve efficient object detection and localization. In natural language processing tasks, they can process sequence data, achieving natural language understanding and so on.

The two most important layers in a CNN are the convolutional layer and the pooling layer. The former is responsible for extracting local features from an image, while the latter down-samples these features. In the process of down-sampling, different pooling methods yield different effects. The earliest pooling methods are max pooling and mean pooling proposed by LeCun and Lin et al. in their early pioneering work [21]. The former preserves the most salient features, while the latter preserves the mean of the overall features, however, both methods require a considerable number of parameters. In 2013, in order to significantly reduce the parameters, Lin et al. introduced global mean pooling in [21], which forced the feature map to correspond to the class and enhanced the interpretability. In 2014, addressing the limitations of fixed pooling strategies, Yu and He put forward a method that enables networks to autonomously learn or select a more suitable pooling strategy [22, 23], while also introducing a fraction to determine the pooling output size [24]. In 2017, to enable networks to find the most suitable pooling function, a parameterized pooling method was proposed, which is learned through back propagation [25, 26]. Since then, researchers have presented a wide array of pooling methods. In 2022, Rodriguez-Martinez et al. used linear combinations of increasing functions in place of traditional pooling methods [27]. Subsequently, (a, b)-grouping functions were employed to replace traditional pooling methods [28]. In 2024, Ferrero–Jaurrieta employed an asymmetric timeout pooling method using pseudo-group functions in convolutional neural networks [29]. Different pooling methods applied in CNNs have all achieved remarkable results.

Some theoretical studies have proved that the more common max pooling in CNNs favors sparser feature representations, but they do not explain why mean pooling performs better in some modern architectures. At the same time, both maximum and mean pooling ignore all possible relationships among the values to be reduced, potentially ignoring important spatial dependencies among the data. Therefore, the choice of pooling aggregation seems to depend on the input data as well as the exact model adopted. In 2018, Dias et al. were the first to propose the use of the Choquet integral for image dimensionality reduction [3, 30]. They demonstrated the effectiveness of the proposed method through the visual results of a photograph and some quantitative results [31]. Following this, investigations were conducted to simulate the effectiveness of the generalized Choquet integral in image dimensionality reduction [30]. The above studies did not consider the modeling of correlations among features, nor did they apply this method to the CNN image classification task. In view of the shortcomings of the existing research, it is very necessary to apply the copula-based Choquet integral pooling method to the CNN image classification task.

In this paper, we propose the Choquet integral pooling method and evaluate it on the Canadian Institute For Advanced Research 10 (CIFAR-10), followed by a comparative analysis with traditional pooling functions. The research demonstrates that the Choquet integral pooling method employs feature maps and convolution kernels in the convolutional layers and pooling layers that differ significantly from those used in traditional pooling methods. Additionally, when randomly sampling images from the database for classification prediction, it was observed that in cases where the spatial relationships within the predicted images were complex and featured interconnections, the classification outcomes of the Choquet integral pooling method were not only superior but also more stable compared to those of conventional pooling techniques.

The paper is organized as follows. In Section 2, we introduce some basic concepts, including the definition and theorem of aggregation functions and the Choquet integral. In Section 3, we study some properties of the Choquet integral based on copulas, providing a corresponding proof and specific computational example. In Section 4, the copula is selected to be the product operator, specifically, $C(x, y) = xy$. This function is then incorporated into the pooling layer of the CNN, and a comprehensive comparative analysis is conducted against traditional pooling methods in terms of convolution kernels, feature maps, and predicted classification outcomes. Our conclusions are presented in Section 5.

2. Preliminaries

First of all, in this section, we use the definition of fuzzy measure for the necessary description and introduction.

Definition 2.1 [32]. A function $A : [0, 1]^n \to [0, 1]$ is an aggregation function whenever the following conditions hold:

(A1) A is increasing in each argument: for each $i \in (1, 2, \cdots n)$, if $x_{(i)} \leq y$, then
$A(x_{(1)}, x_{(2)}, \cdots, x_{(n)}) \leq A(x_{(1)}, x_{(2)}, \cdots, x_{(i-1)}, y, x_{(i+1)}, \cdots, x_{(n)})$.

(A2) A satisfies the boundary conditions: $A(0, \cdots 0) = 0$ and $A(1, \cdots 1) = 1$.

Definition 2.2 [33]. An aggregation function T: $[0, 1]^2 \to [0, 1]$ is a triangular norm ("t-norm" for short), if for all $x, y, z \in [0, 1]$, it satisfies the following properties:

(T1) Commutativity: $T(x, y) = T(y, x)$.

(T2) Associativity: $T(x, T(y, z)) = T(T(x, y), z)$.

(T3) Monotonicity: $T(x, y) \leq T(x, z)$, whenever $y \leq z$.

(T4) Boundary condition: $T(x, 1) = x$.

If T satisfies **(T4)** (and also $T(1, x) = x$ only), then it is called a semi-copula.

Definition 2.3 [34]. A function O: $[0, 1]^2 \to [0, 1]$ is said to be an overlap function if it satisfies the following conditions:

(O1) O is commutative.

(O2) $O(x, y) = 0$ if and only if $xy = 0$.

(O3) $O(x, y) = 1$ if and only if $xy = 1$.

(O4) O is increasing.

(O5) O is continuous.

Definition 2.4 [34]. A bivariate function C: $[0, 1]^2 \to [0, 1]$ is a copula if it satisfies the following conditions, for all $x, x', y, y' \in [0, 1]$ with $x \leq x'$ and $y \leq y'$:

(C1) $C(x, y) + C(x', y') \geq C(x, y') + C(x', y)$.

(C2) $C(x, 0) = C(0, x) = 0$.

(C3) $C(x, 1) = C(1, x) = x$.

Copulas are functions that link (two-dimensional) probability distribution functions to their one-dimensional margins, playing a significant role in the theory of probabilistic metric spaces and statistics [35].

Proposition 2.1 [33, 36]. For each copula C: $[0, 1]^2 \to [0, 1]$, and the Lukasiewicz and minimum T-norms (T_L, T_M) T: $[0, 1]^2 \to [0, 1]$, it holds that:

(1) $T_L \leq C \leq T_M$.

(2) C is increasing.

(3) C satisfies the Lipschitz property with constant 1, that is, for all x_1, x_2, y_1, $y_2 \in [0, 1]$, one has that:

$$|C(x_1, y_1) - C(x_2, y_2)| \leq |x_1 - x_2| + |y_1 - y_2|.$$

An immediate consequence of Proposition 2.1 is that any copula is continuous. Then, each associative copula is a continuous t-norm [33].

Theorem 2.1 [33]. Let T: $[0, 1]^2 \to [0, 1]$ be a t-norm. Then, the following statements are equivalent:

(1) T is a copula.

(2) T satisfies the Lipschitz property with constant 1.

The definitions of overlap, t-norms and copulas can be easily extended to n-ary functions [33, 37–39].

Definition 2.5 [40]. Let $\overrightarrow{r} = (r_1, \cdots, r_n)$ be a real n-dimensional vector, $\overrightarrow{r} \neq \overrightarrow{0} = (0, \cdots, 0)$, and a function F: $[0, 1]^2 \to [0, 1]$ is said to be $\overrightarrow{r}$-increasing, if for all $\overrightarrow{x} = (x_1, \cdots, x_n) \in [0, 1]^n$ and for every $c > 0$, such that $\overrightarrow{x} + c\overrightarrow{r} = (x_1 + cr_1, \cdots, x_n + cr_n) \in [0, 1]^n$, it holds that

$$F(\overrightarrow{x} + c\overrightarrow{r}) \geq F(\overrightarrow{x}).$$

Similarly, one defines an $\overrightarrow{r}$-decreasing function.

In Definition 2.4, when $\overrightarrow{r} = (1, \cdots, 1) = \overrightarrow{1}$, the $\overrightarrow{r}$-increasing function F is called weakly increasing [41].

Considering the notion of directional monotonicity, Lucca et al. [14] introduced the concept of a pre-aggregation function.

Definition 2.6 [14]. A function $PA:[0,1]^n \to [0,1]$ is said to be an n-ary pre-aggregation if the following conditions hold:

(PA1) Directional increasingness: There exists $\overrightarrow{r} = (r_1, \cdots, r_n) \in [0,1]^n$, $\overrightarrow{r} \neq \overrightarrow{0}$, such that PA is $\overrightarrow{r}$-increasing.

(PA2) Boundary conditions: (1)$PA(0, \cdots, 0) = 0$ and (2) $PA(1, \cdots, 1) = 1$.

If F is a pre-aggregation function with respect to a vector $\overrightarrow{r}$, we just say that F is an $\overrightarrow{r}$-pre-aggregation function.

By comparing the definitions of the aggregation function and the pre-aggregation function, we find that the requirements of the pre-aggregation function are weaker than those of the aggregation function. The Choquet integral is also an aggregation function in essence, but the generalization of the Choquet integral is not necessarily an aggregation function.

The Choquet integral is the natural generalization of the Lebesgue integral, where the definition of the Lebesgue integral considers the additive measure, while the definition of the Choquet integral considers the fuzzy measure. Considering that $N = (1, 2, \cdots, n)$ is a finite set, the definition of the discrete Choquet integral is introduced on the basis of a fuzzy measure.

Nowadays, we present the concept of a fuzzy measure [42], which is a central tool for defining the Choquet integral. In what follows, denote $N = (1, 2, \cdots, n)$, for an arbitrary $n > 0$.

Definition 2.7 A function m: $2^N \to [0,1]$ is said to be a fuzzy measure if, for all X, $Y \subset N$, it satisfies the following properties:

(m1) Increasing: If $X \subset Y$, then $m(X) \subset m(Y)$.

(m2) Boundary conditions: $m(\varnothing) = 0$ and $m(N) = 1$.

Regarding aggregation functions, we use the fuzzy measure to analyze the relationship among the elements that we are aggregating, obtaining the relevance of a coalition. In this paper, we adopt the power measure m_{PM}: $2^N \to [0,1]$, which is defined, for all $X \subset N$, by

$$m_{PM} = (\tfrac{|X|}{n})^q, \text{ with } q > 0.$$

Definition 2.8 [32]. Let m: $2^N \to [0,1]$ be a fuzzy measure. The discrete Choquet integral is the function $\mathfrak{C}_{\mathfrak{m}}$: $[0,1]^n \to [0,1]$, defined, for all of $\overrightarrow{x} = (x_1, \cdots, x_n) \in [0,1]^n$, by

$$C_m(\overrightarrow{x}) = \sum_{i=1}^{n} (x_{(i)} - x_{(i-1)}) \cdot m(A_{(i)}),$$

where

(1) $(x_{(1)}, \cdots, x_{(n)})$ is an increasing permutation on the input x, that is, $0 \leq x_{(1)} \leq \cdots \leq x_{(n)}$;

(2) $x_{(0)} = 0$;

(3) $A_{(i)} = \{(i), \cdots, (n)\}$ is the subset of indices corresponding to the $n - i + 1$ largest components of $\overrightarrow{x}$.

Observe that the equation can be also written as

$$C_m(\overrightarrow{x}) = \sum_{i=1}^{n} (x_{(i)} \cdot m(A_{(i)}) - x_{(i-1)} \cdot m(A_{(i)})),$$

which we call the Choquet integral in its expanded form.

When using the Choquet integral in order to aggregate the inputs, it is possible to consider the relevance of the different coalitions (groups of inputs). The latest development of the generalization of the standard form of the Choquet integral is mentioned in the literature [14]. Because these generalization methods are very ordinary, we can replace the product operator in Choquet integral with a fusion function with proper properties. The C_C-integral is obtained by substituting Copula for the product operator in the Choquet integral, which is a pre-aggregate function. Its specific form is as follows.

Definition 2.9 [15]. Let m: $2^N \to [0,1]$ be a fuzzy measure and $C : [0,1]^2 \to [0,1]$ be a bivariate copula. The Choquet-like copula-based integral with respect to m is defined as a function $\mathfrak{C}_{\mathfrak{m}}$: $[0,1]^n \to [0,1]$, given, for all $x \in [0,1]^n$, by

$$C_m^C(\overrightarrow{x}) = \sum_{i=1}^{n} (C(x_{(i)}, m(A_{(i)})) - C(x_{(i-1)}, m(A_{(i)}))),$$

where

(1) $(x_{(1)}, \cdots, x_{(n)})$ is an increasing permutation on the input x, that is, $0 \le x_{(1)} \le \cdots \le x_{(n)}$;

(2) $x_{(0)} = 0$;

(3) $A_{(i)} = \{(i), \cdots, (n)\}$ is the subset of indices corresponding to the $n - i + 1$ largest components of $\overrightarrow{x}$.

Among them, when the copula is taken as the triangular norm, the C_C-integral is transformed into the C_T-integral, which is the first generalization of the standard form of the Choquet integral. Also it plays an important role in the process of dealing with classification problems.

Definition 2.10 [15]. Let m: $2^N \to [0,1]$ be a fuzzy measure and $C : [0,1]^2 \to [0,1]$ be a t-norm. Taking as basis the Choquet integral, a C_T-integral with respect to m is defined as a function $\mathfrak{C}_{\mathfrak{m}}$: $[0,1]^n \to [0,1]$, given, for all $x \in [0,1]^n$, by

$$C_m^C(\overrightarrow{x}) = \sum_{i=1}^{n} T(x_{(i)} - x_{(i-1)}, m(A_{(i)})),$$

where

(1) $(x_{(1)}, \cdots, x_{(n)})$ is an increasing permutation on the input x, that is, $0 \le x_{(1)} \le \cdots \le x_{(n)}$;

(2) $x_{(0)} = 0$;

(3) $A_{(i)} = \{(i), \cdots, (n)\}$ is the subset of indices corresponding to the $n - i + 1$ largest components of $\overrightarrow{x}$.

3. Some properties of the Choquet integral based on copulas

This section presents a novel proof method for the properties of the C_C-integral, as well as a new proof for a known theorem introduced in [15]. Unlike the method in [15], which depends on specific algebraic properties of copulas such as associativity and monotonicity, our proofs adopt structured recursion and low-to-high-dimensional induction, and are further grounded in the probabilistic interpretation of copula and fuzzy measure axioms. The proposed approach simplifies the reasoning process, relies on the fundamental nature of copulas, avoids complicated integral techniques and measure operations in the original proof, and enhances both the generality and robustness of the derivation, while providing deeper theoretical insights.

Theorem 3.1 Let $C : [0,1]^2 \to [0,1]$ be a copula such that $C(x,y) \le x$ for every $x, y \in [0,1]$. Then

$$C_m^C(x_{(1)}, x_{(2)}, \cdots, x_{(n)}) \le max(x_{(1)}, x_{(2)}, \cdots, x_{(n)}),$$

for every $(x_{(1)}, x_{(2)}, \cdots, x_{(n)}) \in [0,1]^n$.

Proof. We prove the result by mathematical induction.

Due to $C(0,y) = 0$ and $C(x,y) \le x$, for $n = 1$,

$$\begin{aligned} C_m^C(x_{(1)}) &= (C(x_{(1)}, m(A_{(1)})) - C(x_{(0)}, m(A_{(1)}))) \\ &= C(x_{(1)}, m(A_{(1)})) - 0 \\ &\le x_{(1)}. \end{aligned}$$

Assume for any $n = k \ge 1$, the following results hold on:

$$C_m^C(x_{(1)}, x_{(2)}, \cdots, x_{(k)}) \le max(x_{(1)}, x_{(2)}, \cdots, x_{(k)}) = x_{(k)}.$$

Consider for any $n = k+1$, and $m(A_{(k+1)}) = 1$,

$$\begin{aligned} C_m^C(x_{(1)}, x_{(2)}, \cdots, x_{(k)}, x_{(k+1)}) &= \sum_{i=1}^{k+1} (C(x_{(i)}, m(A_{(i)})) - C(x_{(i-1)}, m(A_{(i)}))) \\ &= \sum_{i=1}^{k} [C(x_{(i)}, m(A_{(i)})) - C(x_{(i-1)}, m(A_{(i)}))] \\ &+ [C(x_{(k+1)}, m(A_{(k+1)})) - C(x_{(k)}, m(A_{(k+1)}))] \\ &\le x_{(k)} + x_{(k+1)} - x_{(k)} \\ &= x_{(k+1)} \\ &= max(x_{(1)}, x_{(2)}, \cdots, x_{(k)}, x_{(k+1)}). \end{aligned}$$

Thus, the statement holds for $n = k+1$.

The proof of the theorem is now completed by using the mathematical induction.

Theorem 3.2 Let $C : [0,1]^2 \to [0,1]$ be a copula such that $C(x,1) = x$ for every $x, y \in [0,1]$. Then

$$C_m^C(x_{(1)}, x_{(2)}, \cdots, x_{(n)}) \ge min(x_{(1)}, x_{(2)}, \cdots, x_{(n)}),$$

for every $(x_{(1)}, x_{(2)}, \cdots, x_{(n)}) \in [0,1]^n$.

Proof. We prove the result by mathematical induction.

Due to $C(x,1) = x$, for $n = 1$,

$$\begin{aligned} C_m^C(x_{(1)}) &= (C(x_{(1)}, m(A_{(1)})) - C(x_{(0)}, m(A_{(1)}))) \\ &= C(x_{(1)}, 1) - C(x_{(0)}1) \\ &= x_{(1)} - x_{(0)} \\ &= x_{(1)}. \end{aligned}$$

Assume for any $n = k \ge 1$, the following results hold on:

$$C_m^C(x_{(1)}, x_{(2)}, \cdots, x_{(k)}) \ge min(x_{(1)}, x_{(2)}, \cdots, x_{(k)}) = x_{(k)}.$$

Consider for any $n = k + 1$, and $m(A_{(k+1)}) = 1$,

$$\begin{aligned} C_m^C(x_{(1)}, x_{(2)}, \cdots, x_{(k)}, x_{(k+1)}) &= \sum_{i=1}^{k+1} (C(x_{(i)}, m(A_{(i)})) - C(x_{(i-1)}, m(A_{(i)}))) \\ &= \sum_{i=1}^{k} [C(x_{(i)}, m(A_{(i)})) - C(x_{(i-1)}, m(A_{(i)}))] \\ &+ [C(x_{(k+1)}, m(A_{(k+1)})) - C(x_{(k)}, m(A_{(k+1)}))] \\ &\geq min(x_{(1)}, x_{(2)}, \cdots, x_{(k)}) + x_{(k+1)} - x_{(k)} \\ &= min(x_{(1)}, x_{(2)}, \cdots, x_{(k)}, x_{(k+1)}), \end{aligned}$$

where due to the definition of the C_C-integral, $x_{(k+1)} - x_{(k)} > 0$, and thus, the statement holds for $n = k+1$.

The proof of the theorem is now completed by using mathematical induction.

Corollary 3.1 According to Theorems 3.1 and 3.2, when $C : [0,1]^2 \to [0,1]$, $C(x,y) \leq x$, and $C(x,1) = x$, we have $min(x_{(1)}, x_{(2)}, \cdots, x_{(n)}) \leq C_m^C(x_{(1)}, x_{(2)}, \cdots, x_{(n)}) \leq max(x_{(1)}, x_{(2)}, \cdots, x_{(n)})$. Thus the C_C-integral is average.

Example 3.1 Let $X = \{x_1, x_2, x_3\}$, $\mathcal{F} = \mathcal{P}(X)$, and the monotone measure μ on $\mathcal{P}(X)$ be given as

$$\mu(E) = \begin{cases} 0 & if \quad E = \emptyset \\ 0.5 & if \quad E = \{x_1\} \\ 0.2 & if \quad E = \{x_2\} \\ 0.6 & if \quad E = \{x_1, x_2\} \\ 0.4 & if \quad E = \{x_3\} \\ 0.7 & if \quad E = \{x_1, x_3\} \\ 0.9 & if \quad E = \{x_2, x_3\} \\ 1 & if \quad E = X. \end{cases}$$

Then (X,$\mathcal{P}(X)$,μ) is a monotone measure space. Let function g: $X \longrightarrow [0, \infty)$ be given as

$$g(x) = \begin{cases} \frac{1}{2} & if \quad x = x_1 \\ \frac{3}{4} & if \quad x = x_2 \\ \frac{1}{4} & if \quad x = x_3, \end{cases}$$

where the copula is $C(x,y) = \min(x,y) \cdot \max(x^2, y^2)$, and thus, $x_1^* = x_3$, $x_2^* = x_1$, $x_3^* = x_2$ such that $g(x_1^*) \leq g(x_2^*) \leq g(x_3^*)$ for $g(x_1^*) = g(x_3) = \frac{1}{4}$ and $\mu(x_1^*, x_2^*, x_3^*) = \mu(X) = 1$. Then, $C(g(x_1^*), \mu(X)) = C(\frac{1}{4}, 1) = \min(\frac{1}{4}, 1) \cdot \max(\frac{1}{16}, 1) = \frac{1}{4} \times 1 = \frac{1}{4} = 0.25$, $C(g(x_0), \mu(X)) = C(0,1) = \min(0,1) \cdot \max(0,1) = 0 \times 1 = 0$, $C(g(x_1^*), \mu(X)) - C(g(x_0), \mu(X)) = \frac{1}{4} = 0.25$ for $g(x_2^*) = g(x_1) = \frac{1}{2}$ and $\mu(x_2^*, x_3^*) = \mu(x_1, x_2) = 0.6$. Then, $C(g(x_2^*), \mu(x_2^*, x_3^*)) = C(\frac{1}{2}, 0.6) = \min(\frac{1}{2}, 0.6) \cdot \max(\frac{1}{4}, 0.36) = \frac{1}{2} \times 0.36 = 0.18$, $C(g(x_1^*), \mu(x_2^*, x_3^*)) = C(\frac{1}{4}, 0.6) = \min(\frac{1}{4}, 0.6) \cdot \max(\frac{1}{16}, 0.36) = \frac{1}{4} \times 0.36 = 0.09$,

$C(g(x_2^*), \mu(x_2^*, x_3^*)) - C(g(x_1^*), \mu(x_2^*, x_3^*)) = 0.09$ for $g(x_3^*) = g(x_2) = \frac{3}{4}$, and $\mu(x_3^*) = \mu(x_2) = 0.2$. Then, $C(g(x_3^*), \mu(x_3^*)) = C(\frac{3}{4}, 0.2) = \min(\frac{3}{4}, 0.2) \cdot \max(\frac{9}{16}, 0.04) = 0.2 \times \frac{9}{16} = 0.1125$, $C(g(x_2^*), \mu(x_3^*)) = C(\frac{1}{2}, 0.2) = \min(\frac{1}{2}, 0.2) \cdot \max(\frac{1}{4}, 0.04) = 0.2 \times \frac{1}{4} = 0.05$, $C(g(x_3^*), \mu(x_3^*)) - C(g(x_2^*), \mu(x_3^*)) = 0.0625$, so the C_C-integral of g with respect to μ (on X) can be calculated as

$$\begin{aligned} C_m^C(g(x_1), g(x_2), g(x_3)) &= \sum_{i=1}^{3} (C(g(x_{(i)}), m(E_{(i)})) - C(g(x_{(i-1)}), m(E_{(i)}))) \\ &= [C(g(x_1^*), \mu(X)) - C(g(x_0), \mu(X))] + [C(g(x_2^*), \mu(x_2^*, x_3^*)) - C(g(x_1^*), \mu(x_2^*, x_3^*))] \\ &+ [C(g(x_3^*), \mu(x_3^*)) - C(g(x_2^*), \mu(x_3^*))] \\ &= 0.25 + 0.09 + 0.0625 \\ &= 0.4025. \end{aligned}$$

Furthermore, $\min(g(x_1), g(x_2), g(x_3)) = \min(\frac{1}{2}, \frac{3}{4}, \frac{1}{4}) = \frac{1}{4}$, $\max(g(x_1), g(x_2), g(x_3)) = \max(\frac{1}{2}, \frac{3}{4}, \frac{1}{4}) = \frac{3}{4}$, and therefore,

$$\begin{aligned} \min(g(x_1), g(x_2), g(x_3)) &= \min(\frac{1}{2}, \frac{3}{4}, \frac{1}{4}) = \frac{1}{4} \\ &\le C_m^C(g(x_1), g(x_2), g(x_3)) = 0.4025 \\ &\le \max(g(x_1), g(x_2), g(x_3)) = \frac{3}{4}. \end{aligned}$$

Thus, the C_C-integral is average.

Theorem 3.3 Let $C : [0,1]^2 \to [0,1]$ be a copula such that $C(x,1) = x$ and $C(0,y) = 0$ for every $x, y \in [0,1]$. Then the C_C-integral is idempotent.

Proof. We prove the result by mathematical induction.

Due to $C(0,y) = 0$ and $C(x,1) = x$, for $n = 1$, $m(A_{(1)}) = 1$,

$$\begin{aligned} C_m^C(x_{(1)}) &= (C(x_{(1)}, m(A_{(1)})) - C(x_{(0)}, m(A_{(1)}))) \\ &= C(x_{(1)}, m(A_{(1)})) - 0 \\ &= C(x_{(1)}, 1) \\ &= x_{(1)}. \end{aligned}$$

Assume for any $n-1$, that the following result hold:

$$C_m^C(x_{(1)}, x_{(2)}, \cdots, x_{(n-1)}) = \sum_{i=1}^{n-1} [C(x_{(i)}, m(A_{(i)})) - C(x_{(i-1)}, m(A_{(i)}))] = x.$$

Consider, for any n, $m(A_{(n)}) = 1$, and $x_{(1)} = x_{(2)} = \cdots = x_{(n-1)} = x_{(n)}$,

$$\begin{aligned} C_m^C(x_{(1)}, x_{(2)}, \cdots, x_{(n-1)}, x_{(n)}) &= \sum_{i=1}^{n} (C(x_{(i)}, m(A_{(i)})) - C(x_{(i-1)}, m(A_{(i)}))) \\ &= \sum_{i=1}^{n-1} [C(x_{(i)}, m(A_{(i)})) - C(x_{(i-1)}, m(A_{(i)}))] \\ &+ [C(x_{(n)}, m(A_{(n)})) - C(x_{(n-1)}, m(A_{(n)}))] \\ &= x + C(x_{(n)}, 1) - C(x_{(n-1)}, 1) \\ &= x + 0 \\ &= x. \end{aligned}$$

Thus, the statement holds for n.

The proof of the theorem is now completed by using mathematical induction.

Example 3.2 Let $X = \{x_1, x_2, x_3\}$, $\mathcal{F} = \mathcal{P}(X)$, and the monotone measure μ on $\mathcal{P}(X)$ be given as

$$\mu(E) = \begin{cases} 0 & if \quad E = \emptyset \\ 0.4 & if \quad E = \{x_1\} \\ 0.1 & if \quad E = \{x_2\} \\ 0.5 & if \quad E = \{x_1, x_2\} \\ 0.3 & if \quad E = \{x_3\} \\ 0.6 & if \quad E = \{x_1, x_3\} \\ 0.8 & if \quad E = \{x_2, x_3\} \\ 1 & if \quad E = X. \end{cases}$$

Then $(X,\mathcal{P}(X),\mu)$ is a monotone measure space. Let function g: $X \longrightarrow [0, \infty)$ be given as

$$g(x) = \begin{cases} \frac{1}{2} & if \quad x = x_1 \\ \frac{3}{4} & if \quad x = x_2 \\ \frac{1}{4} & if \quad x = x_3, \end{cases}$$

where copula is $C(x, y) = \min(x, y) \cdot \max(x^2, y^2)$.

Due to $x, y \in [0, 1]$, thus $C(x, 1) = \min(x, 1) \cdot \max(x^2, 1) = x \times 1 = x$, $C(x, 0) = \min(x, 0) \cdot \max(x^2, 0) = 0 \times x^2 = 0$. Thus, $x_1^* = x_3$, $x_2^* = x_1$, $x_3^* = x_2$ such that $g(x_1^*) \leq g(x_2^*) \leq g(x_3^*)$ for $g(x_1^*) = g(x_3) = \frac{1}{4}$ and $\mu(x_1^*, x_2^*, x_3^*) = \mu(X) = 1$. Then, $C(g(x_1^*), \mu(X)) = C(\frac{1}{4}, 1) = \min(\frac{1}{4}, 1) \cdot \max(\frac{1}{16}, 1) = \frac{1}{4} \times 1 = \frac{1}{4} = 0.25$, $C(g(x_0), \mu(X)) = C(0, 1) = \min(0, 1) \cdot \max(0, 1) = 0 \times 1 = 0$,

$C(g(x_1^*), \mu(X)) - C(g(x_0), \mu(X)) = \dfrac{1}{4} = 0.25$, Therefore,

$$\begin{aligned}
C_m^C(g(x_1^*), g(x_1^*), g(x_1^*)) &= \sum_{i=1}^{3}(C(g(x_{(i)}), m(E_{(i)})) - C(g(x_{(i-1)}), m(E_{(i)})) \\
&= [C(g(x_1^*), \mu(X)) - C(g(x_0), \mu(X))] + [C(g(x_1^*), \mu(E_{(2)})) - C(g(x_1^*), \mu(E_{(2)}))] \\
&+ [C(g(x_1^*), \mu(E_{(3)})) - C(g(x_1^*), \mu(E_{(3)}))] \\
&= C(\frac{1}{4}, 1) - C(0, 1) + 0 + 0 \\
&= \{\min(\frac{1}{4}, 1) \cdot \max(\frac{1}{16}, 1)\} - \{\min(0, 1) \cdot \max(0, 1)\} \\
&= (\frac{1}{4} \times 1) - (0 \times 1) \\
&= \frac{1}{4} \\
&= g(x_1^*).
\end{aligned}$$

Similary, for $g(x_2^*)$ and $g(x_3^*)$, the above formula holds true, thus the C_C-integral is idempotent. In particular, when the copula is taken as a triangular norm, the following properties hold,

Theorem 3.4 Let $C \in \aleph$, $n \geq 2$, for every $m \in M$. The C_C-integral is translation-invariant, where $\aleph$ is the set of all the copula functions that satisfy $C(0, y) = 0$, for every $y \in [0, 1]$. The set of all measures $m : 2^n \to [0, 1]$ is **M**.

Proof. If a permutation$(\cdot)$ satisfies $x_{(1)} \leq x_{(2)} \leq \cdots \leq x_{(n)}$, thus $x_{(1)} + c \leq x_{(2)} + c \leq \cdots \leq x_{(n)} + c$, then

$$\begin{aligned}
C_m^C(x_{(1)} + c, x_{(2)} + c, \cdots, x_{(n)} + c) &= C(x_{(1)} + c, 1) + \sum_{i=2}^{n} C(x_{(i)} - x_{(i-1)}, m(A_{(i)})) \\
&= C(x_{(1)} + c, 1) - C(x_{(1)}, 1) + \sum_{i=1}^{n} C(x_{(i)} - x_{(i-1)}, m(A_{(i)})),
\end{aligned}$$

for any $x \in [0, 1]^n, c \in (0, 1]$. So, the translation invariance of the C_C-integral is equivalent to $C(x_{(1)} + c, 1) = C(x_{(1)}, 1) + c$, due to properties of copula functions: $C(x, 1) = x$. Therefore

$$C(x_{(1)} + c, 1) = x_{(1)} + c,$$

and similarly

$$C(x_{(1)}, 1) = x_{(1)}.$$

We have

$$C(x_{(1)} + c, 1) = x_{(1)} + c = C(x_{(1)}, 1) + c,$$

and the C_C-integral is translation-invariant.

Example 3.3 Due to translation-invariance of the C_C-integral is equivalent to $C(x_{(1)}+c, 1) = C(x_{(1)}, 1)+c$, thus $x_{(1)} = \dfrac{1}{4}$, $c = \dfrac{1}{2}$, $C(x, y) = \min(x, y)$.

$$C(x_{(1)} + c, 1) = C(\frac{1}{4} + \frac{1}{2}, 1) = \min(\frac{3}{4}, 1) = \frac{3}{4},$$

$$C(x_{(1)}, 1) + c = C(\frac{1}{4}, 1) + \frac{1}{2} = \min(\frac{1}{4}, 1) + \frac{1}{2} = \frac{1}{4} + \frac{1}{2} = \frac{3}{4}.$$

As a result $C(x_{(1)} + c, 1) = x_{(1)} + c = C(x_{(1)}, 1) + c$, and the C_C-integral is translation invariant.

Remark 3.1 It is worthing noting that if the copula function is replaced with another type of function, the aforementioned theorem may no longer be valid. The Choquet integral, as the standard form of the C_C-integral, exhibits the property of translation invariance. This feature becomes particularly significant in the subsequent chapter, where the Choquet integral is employed as a pooling function within CNN.

Review the positive homogeneity of the Choquet integral, that is, $Ch_m(cx) = cCh_m(x)$,for all $c > 0$, $x, cx \in [0, 1]^n$.

Next, we study the positive homogeneity of the C_C-integral. Because we need to consider the properties of copulas, we divide copulas into three classes: t-norms, overlap functions, and non-associative copulas.

When copulas are t-norms, and because t-norms satisfied positive homogeneity, thus we have

Theorem 3.5 [43]. Let $T \in \aleph$, $n \geq 2$, for any $m \in M$. The C_T-integral has positive homogeneity if and only if T is positive homogeneous on the first variable, that is, $T(cx, y) = cT(x, y)$ for all $c > 0$, x, cx, $y \in [0, 1]$.

When copulas are overlap functions, because not all overlap functions satisfy positive homogeneity, then we have

Theorem 3.6 [44]. Let $O : [0.1]^2 \to [0, 1]$ be a homogeneous overlap function generated by a pair of multiplicative geneeators. Then, for all $z, y \in [0, 1]$, the function satisfies $O^p(z, y) = z^p \cdot y^p$, where $p > 0$ is a constant parameter.

Corollary 3.2 The C_O-integral has positive homogeneity if and only if O is positive homogeneous on the first variable, that is, $O(z, y) = z \cdot y$.

When copulas are non-associative copulas, they do not have positive homogeneity, thus the C_C-integral does not have positive homogeneity.

Example 3.4 For non-associative copulas, take $C(x, y) = xy + x^2y(1 - x)(1 - y)$ as an example, for all $c > 0$, x, cx, $y \in [0, 1]$, due to

$$C(cx, y) = (cx)y + (cx)^2y(1 - cx)(1 - y) = cxy + c^2x^2y(1 - cx)1 - y),$$

$$c \cdot C(x, y) = c \cdot (xy + x^2y(1 - x)(1 - y)) = cxy + cx^2y(1 - x)(1 - y),$$

obviously, $C(cx, y) \neq c \cdot C(x, y)$. Therefore, they do not have positive homogeneity, thus the C_C-integral does not have positive homogeneity.

In summary, the conditions for the validity of C_C-integral properties are more stringent, requiring consideration of both Choquet integral properties and copula properties. Particularly, when the copula function is $C(x, y) = xy$, the C_C-integral is a standard Choquet integral. Therefore, in the following, we apply the Choquet integral to the pooling layer in the CNN to form a Choquet integral pooling layer.

4. The application of the Choquet integral in the convolutional neural network

4.1. Traditional convolutional neural networks

A CNN is a deep learning model that extracts features with good performance from input data by simulating the processing mode of the human visual system, so as to extract features and recognize patterns efficiently. It is widely used in image recognition, video analysis, natural language processing, and many other fields. In addition, CNNs are an extension of traditional neural networks designed to process data grid topologies, such as images (2D grids) and videos (3D grids).

The basic structure of a CNN includes an input layer, convolution layer, pooling layer, full connection layer, and so on. The convolution layer is the core of the CNN, which extracts local features of input data by a convolution operation. During the convolution operation, the convolution kernel slides on the input data, and makes weighted summation on the local region to obtain the new feature map. This process simulates the receptive field mechanism of neurons in the human visual system, which enables the CNN to learn the spatial structure information and feature representation of the image automatically. The pooling layer is used to sample the feature graph of the convolution layer to reduce the data dimension and computation, and to enhance the robustness of the model. Normal pooling operations include maximum pooling and mean pooling. Maximum pooling can extract the most significant features in local areas, while mean pooling pays more attention to the preservation of overall information. The fully connected layer usually at the end of a CNN is used to integrate features extracted from the previous layer and output the final classification or regression results. During training, the CNN learns the mapping between input data and output tags by backpropagating the parameters of each layer to minimize the loss function.

This article focuses on the pooling layer. As a key component of a CNN, the primary function of the pooling layer is to down-sample the output feature maps from convolutional layers. This process is designed to reduce the dimensionality of the feature maps, thereby decreasing the number of parameters and the computational complexity of the model, which improves computational efficiency. The pooling layer is typically applied after the convolutional layer. It effectively extracts the dominant features from the input data, eliminates redundant information, thereby forcing the model to concentrate on learning essential characteristics. Concurrently, it enhances the generalization capability of the model and reduces the risk of overfitting. Owing to the translation invariance of the pooling operation, that is, when the input data is slightly translated, the output of the pooling layer is basically unchanged, which makes the model more robust to small changes in the input data. This is analogous to the translation invariance of the Choquet integral based on copulas studied in the previous section.

We take color classification in imaging as an example, that is, the CIFAR-10 database. It is an open-source handwritten digital database of 60,000 images, 50,000 of which are used for training the CNN and 10,000 for testing the CNN. We divide this 50,000 training data into two parts, one part as training data and one part as check data, where the training data is used to model training, the check data is used if the model is overfitting and to tune parameters, while the test data is used to evaluate the overall performance of the model. Each image is a 32 * 32 pixel image. In addition, the CNN structure defined in this example consists of two convolution layers, two pooling layers, two full connection layers, a ReLU activation function, and a dropout layer. It first passes through the initial convolutional layer and ReLU activation function, then goes through a pooling layer. Subsequently, the output from the first pooling layer passes through the second convolutional layer and ReLU activation function,

followed by the second pooling layer. After this, the data is flattened and processed through two fully connected layers, finally generating classification probabilities through the Softmax layer. The specific network structure diagram is shown in Figure 1. The parameters presented in the methodology were configured as follows: stride $s = 2$, fuzzy measure exponent $q = 2$, window size $k = 2 \times 2$. These values were chosen by the specialists in order to obtain better performance of the function and based on a pilot experiment using the same methodology [30] . In the pooling layer, we use the maximum pooling and mean pooling modes. In the following sections, we will analyze the results of these two kinds of pooling, as well as the variation of the characteristic graph and the characteristic core of the convolution layer during the CNN process. We take the classification prediction of 36 images using a 20-cycle image as an example. Algorithm 1 presents the corresponding simplified calculation process.

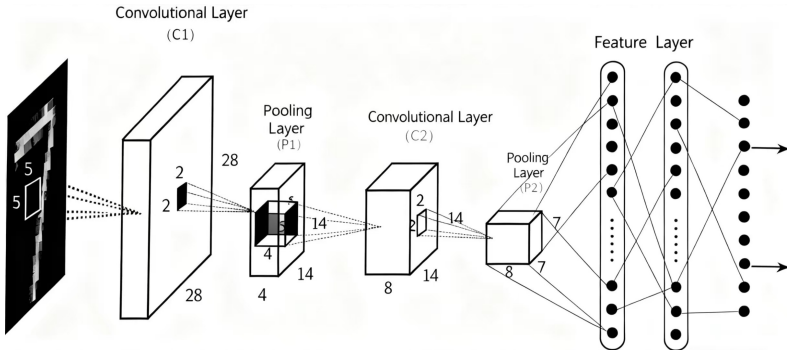

Figure 1. The structure of the convolutional neural network.

Algorithm 1 The algorithm for the simplified calculation process

Input: Image: $I \in \mathbb{R}^{H \times W}$, window size: $k = 2 \times 2$, stride $s = 2$, fuzzy measure exponent $q = 2$.
Output: Pooling feature map: $O \in \mathbb{R}^{H' \times W'}$

$O \leftarrow \text{zeros}(H', W')$, $H' = \lfloor \frac{H-k}{s} \rfloor + 1$, $W' = \lfloor \frac{W-k}{s} \rfloor + 1$
$I_{norm} \leftarrow I/255$
for $i' = 0$ to $H' - 1$ **do**
  for $j' = 0$ to $W' - 1$ **do**
    $W_{norm} \leftarrow I_{norm}[i's : i's + k, j's : j's + k]$
    $x \leftarrow \text{sort}(\text{flatten}(W_{norm}))$
    $x_{(0)} \leftarrow 0$
    $C_\mu \leftarrow 0$
    for $i = 1$ to k^2 **do**
      $d_i \leftarrow x_{(i)} - x_{(i-1)}$
      $\mu_i \leftarrow \left(\frac{k^2-i+1}{k^2}\right)^q$
      $C_\mu \leftarrow C_\mu + d_i \cdot \mu_i$
    end for
    $O[i', j'] \leftarrow C_\mu \times 255$
  end for
end for
return O

4.1.1. Max pooling

The maximum pooling function is the selection of the largest element in the defined spatial neighborhood as the output. This method has been proven to be effective in a variety of tasks. It can highlight the salient information in the feature map, and is especially effective for capturing details such as texture and edge.

After 20 cycles of training, its accuracy rate is shown in Table 1. It can be clearly seen from the table that with the increase of training rounds, the accuracy rate of the maximum pooling image recognition has shown a steady upward trend, and finally reaches a higher level of 98.60%. Nevertheless, it does not imply a high accuracy rate in image classification tasks. This indicates that max pooling can effectively capture the significant information in images for correct classification. At the same time, it shows that the model not only remembers the training data but also makes correct predictions for unseen images (such as samples under different angles and lighting conditions), demonstrating a certain degree of generalization ability.

Table 1. Maximum pooling accuracy.

Epoch	Loss	Training accuracy	Test accuracy
20 (10%)	0.013838	98.9%	98.44%
20 (20%)	0.008843	98.89%	98.36%
20 (30%)	0.014151	98.97%	98.42%
20 (40%)	0.010357	98.98%	97.42%
20 (50%)	0.004465	98.99%	98.46%
20 (60%)	0.004991	99%	98.6%
20 (70%)	0.027664	99.02%	98.52%
20 (80%)	0.025188	99.04%	98.58%
20 (90%)	0.006584	99.06%	98.38%
20 (100%)	0.061886	96.88%	98.4%

The convolution operation is performed before the first pooling operation. The first convolutional layer performs initial recognition by extracting the edge textures of the image, where the convolution kernel and the characteristic diagram of the first convolution layer are shown in Figures 2 and 3.

In order to balance the depth of feature extraction and the computational complexity, the convolution and pooling operations of the second layer are performed after the first layer of pooling operation. The convolution cores and characteristic graphs of the second convolution layer are shown in Figures 4 and 5. It is evident from the graphic that only the important features are illuminated in the feature map of the maximum pooling method with many other areas being black. As can be seen, there is a lot of contrast in the feature map.

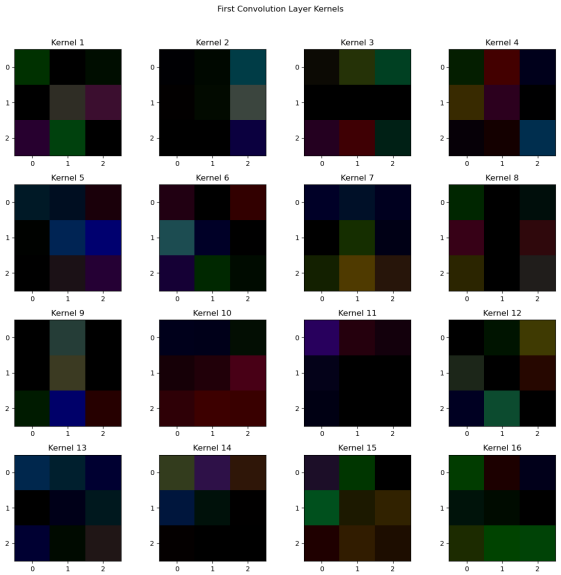

Figure 2. The convolution kernel of the first convolution layer.

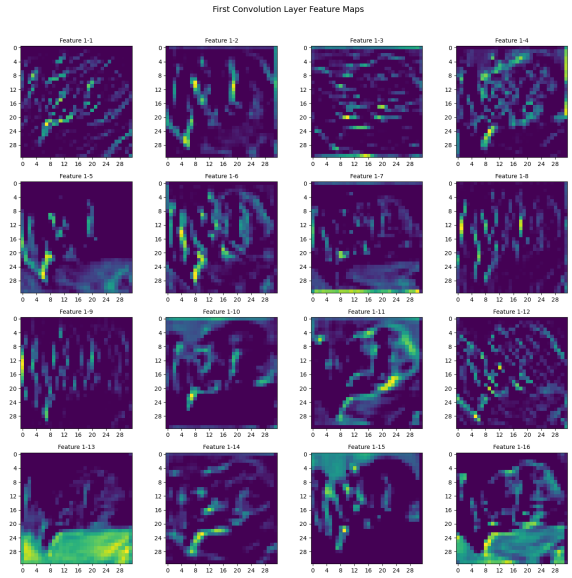

Figure 3. The characteristic diagram of the first convolution layer.

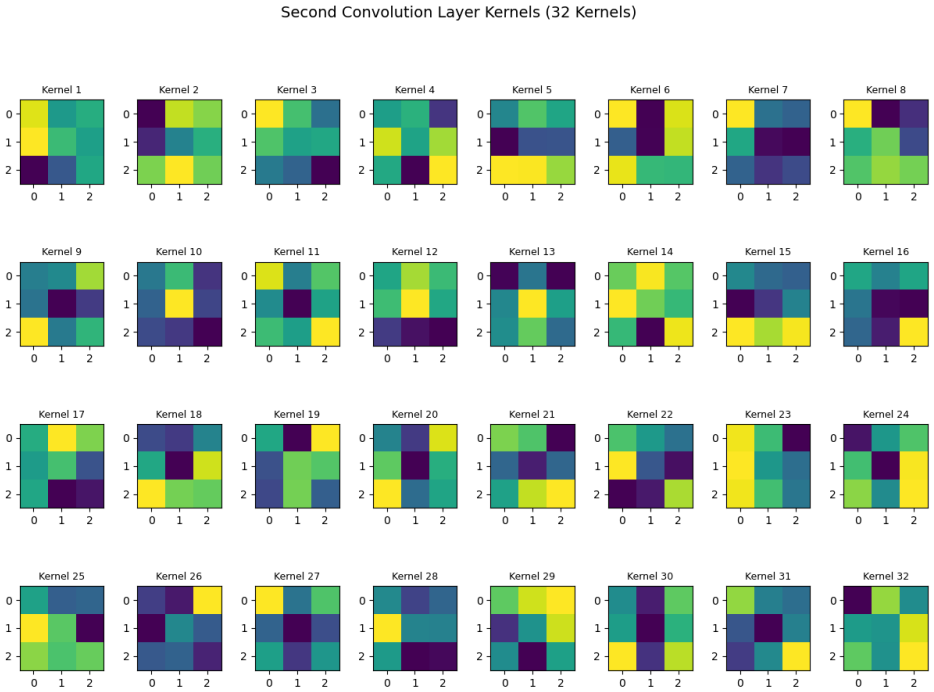

Figure 4. The convolution kernel of the second convolution layer.

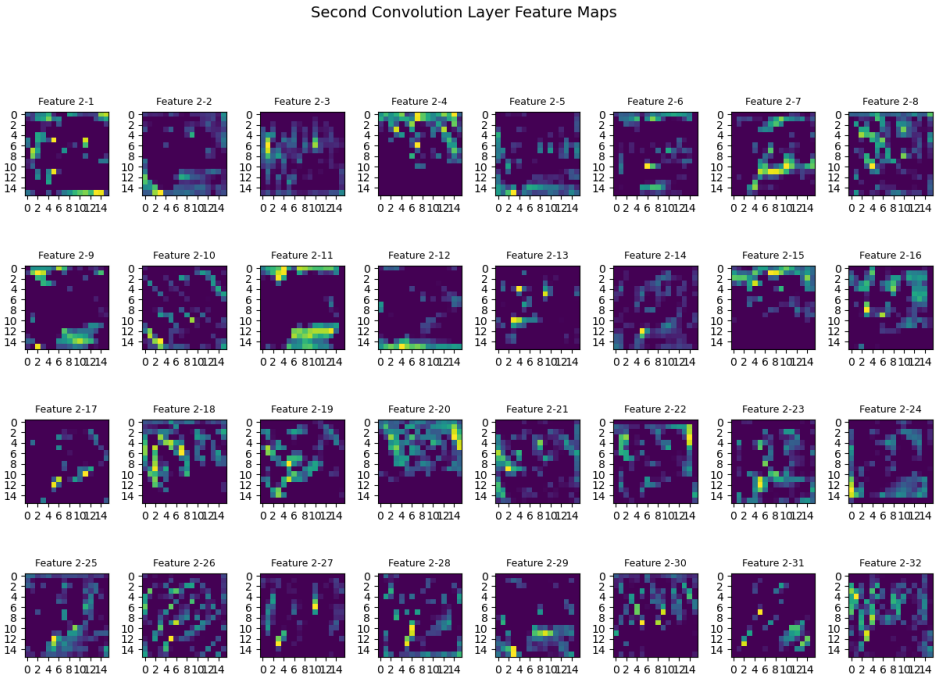

Figure 5. The characteristic diagram of the second convolution layer.

Through the analysis of the pooling layer and convolution layer, it is found that maximum pooling has played an excellent role in the pooling layer. It makes network training and reasoning more efficient. It can also identify features with small displacements in the input data and reduce the risk of overfitting in the case of limited data. However, in other respects, maximization has its limitations. It may lose some important information because it only retains the maximum in a local area and ignores other values, not being able to capture all of the important features, especially when multiple important features coexist within a local area. To overcome some of the limitations of maximization, it may be necessary to make more adjustments to other parts of the network, such as increasing the number of convolutional layers or adjusting the step size. For certain tasks, maximization may not be the best choice, for example, if the task requires accurate spatial information, maximum pooling can destroy this information.

4.1.2. Mean pooling

The mean pooling function is another pooling method. Unlike maximum pooling, mean pooling computes the average value of all elements in the pooling window as output. This method can smooth the noise and details in the image, and keep the more integral and smooth feature information. Mean pooling is more appropriate in some scenarios, such as when you need to focus on the overall trend of the image rather than the local details. Mean pooling also helps to reduce the risk of overfitting because it can suppress the extreme values in the feature map to some extent.

After training, the accuracy of mean pooling is obtained as shown in Table 2. It can be clearly seen from the table that mean pooling shows a steady upward trend with the increase of the training cycle,

and also achieves a high accuracy rate, that is, 97.74%. Similarly, this does not represent its accuracy rate in the image classification task. The experimental data show that mean pooling does not bring better results than maximum pooling, but the accuracy of the two methods is not much different, so mean pooling also has a good effect on reducing the loss function and the fitting ability of the model is improved effectively.

Table 2. Mean pooling accuracy.

Epoch	Loss	Training accuracy	Test accuracy
20 (10%)	0.077772	98.1%	97.7%
20 (20%)	0.078991	98.1%	97.74%
20 (30%)	0.085286	98.05%	97.54%
20 (40%)	0.142854	98.02%	97.32%
20 (50%)	0.098920	98.01%	97.54%
20 (60%)	0.098299	97.98%	97.64%
20 (70%)	0.021822	98.01%	97.46%
20 (80%)	0.064955	98.02%	97.56%
20 (90%)	0.082679	97.99%	97.6%
20 (100%)	0.046325	98.13%	97.68%

Like maximum pooling, mean pooling in the CNN consists of two convolution layers, two pooling layers, and two fully connected layers. The convolution cores and characteristic maps of the first convolution layer are shown in Figures 6 and 7, and the convolution cores and characteristic maps of the second layer are shown in Figures 8 and 9. As the graphic illustrates, the feature maps of the mean pooling method have a more consistent information distribution that reflects the overall intensity level of the image. Additionally, the contrast of the feature maps is comparatively low.

Through the analysis of the convolution layer and pooling layer of mean pooling, it is discovered that the accuracy of mean pooling is slightly lower than that of maximum pooling, but the background information of the image can be preserved by mean pooling in the feature map of the convolution layer. This is useful for some tasks that require background information, and mean pooling is less computationally involved in some cases than maximum pooling because it does not need to find the maximum. At the same time, mean pooling has its disadvantages, for example, it may lead to blurring of image features, because it mixes all the values in the region, and may lose some important details, especially when the region contains a lot of unimportant background information. Finally, mean pooling may also reduce the contrast of the feature map since it averages high and low activation values, resulting in less prominent features.

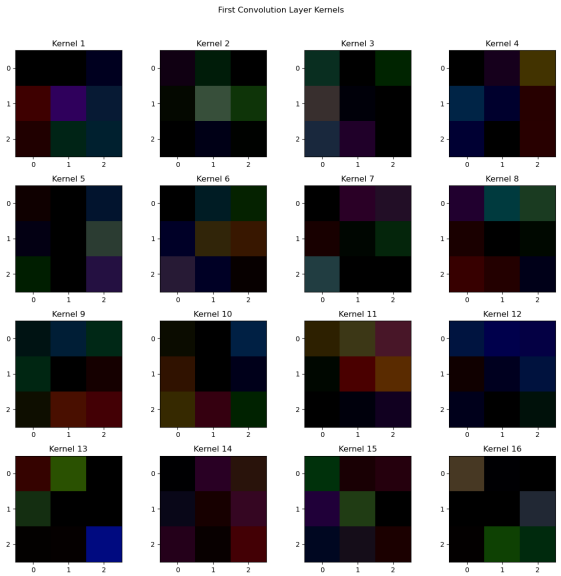

Figure 6. The convolution kernel of the first convolution layer.

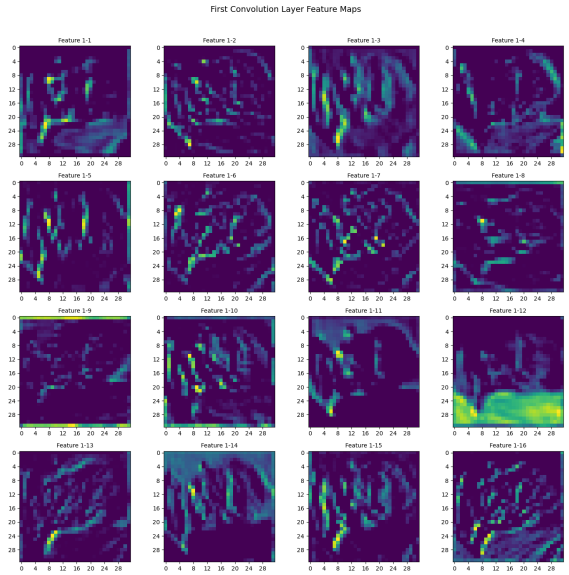

Figure 7. The characteristic diagram of the first convolution layer.

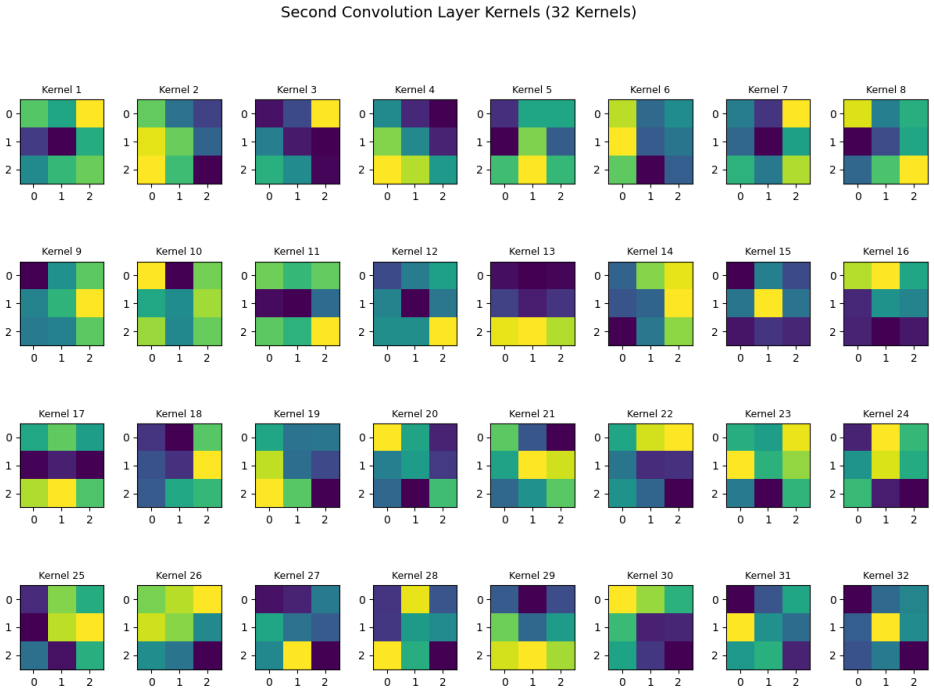

Figure 8. The convolution kernel of the second convolution layer.

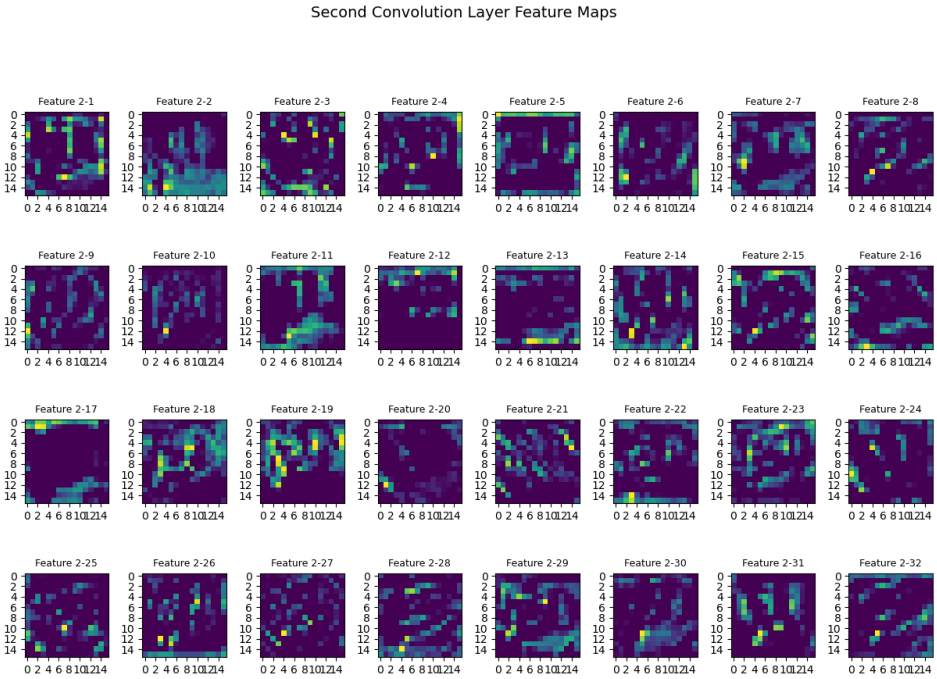

Figure 9. The characteristic diagram of the second convolution layer.

4.2. Improved convolutional neural network

The above analysis of traditional pooling operations shows that both maximum and mean pooling do not take into account the interaction and importance of elements, and there is no way to deal with fuzzy or uncertain information, so a method of pooling based on the fuzzy measure, namely, the Choquet integral pooling method, is proposed. It can be implemented in the CNN as a custom pooling layer that captures the complex structure of data, especially when dealing with data with complex dependencies. This further increases the ability of the model to process complex data. Taking color classification in imaging as an example, the CIFAR-10 dataset was used to analyze the prediction results, and using characteristic cores and maps of the first and second convolution layers in the CNN, the importance and influence of Choquet integral pooling are explained.

After 20 cycles of training, it was detected that the accuracy of the Choquet integral pooling method was increasing, and the highest accuracy was 97.64%. This is not much different from the effect of mean pooling. However, this does not mean that the mean pooling method is superior to the Choquet integral pooling method. A higher recognition accuracy on photographs does not equate to a higher correctness rate in general image classification. In the following section, we will compare the accuracy of the three methods in image classification. The accuracy of the training set and test set of the Choquet integral pooling method is shown in Table 3.

Like the traditional pooling method, the Choquet pooling in the CNN consists of two convolution layers, two pooling layers, and two fully connected layers. The convolution cores and characteristic maps of the first convolution layer are shown in Figures 10 and 11, and the convolution cores and characteristic maps of the second layer are shown in Figures 12 and 13. The Choquet integral pooling

approach has more feature maps, as seen in the figure, which can retain more pertinent information and increase classification accuracy.

Table 3. Choquet pooling accuracy.

Epoch	Loss	Training accuracy	Test accuracy
20 (10%)	0.097973	98.03%	97.4%
20 (20%)	0.175447	98.02%	97.36%
20 (30%)	0.041895	98.03%	97.26%
20 (40%)	0.039211	98.03%	97.38%
20 (50%)	0.007530	98.04%	97.38%
20 (60%)	0.051620	98.04%	97.4%
20 (70%)	0.031938	98.05%	97.64%
20 (80%)	0.083377	98.04%	97.42%
20 (90%)	0.018999	98.05%	97.4%
20 (100%)	0.066430	98.05%	97.32%

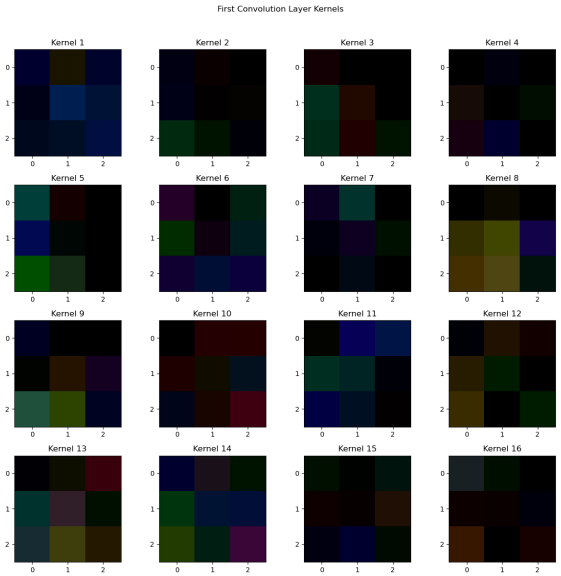

Figure 10. The convolution kernel of the first convolution layer.

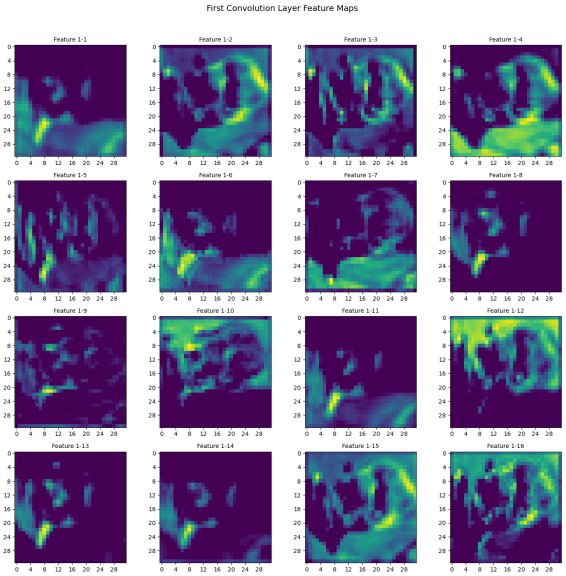

Figure 11. The characteristic diagram of the first convolution layer.

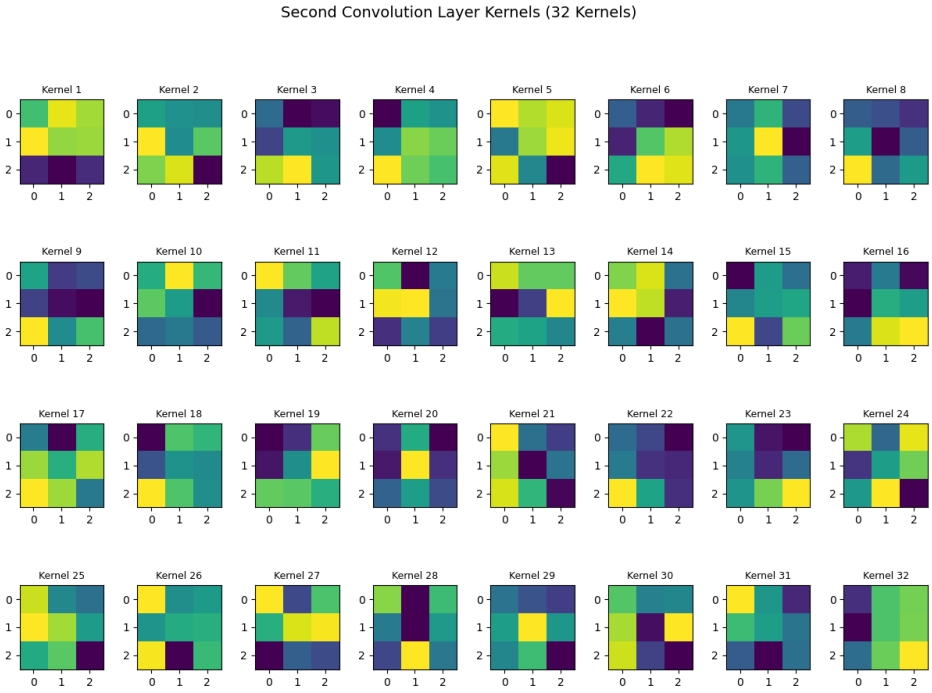

Figure 12. The convolution kernel of the second convolution layer.

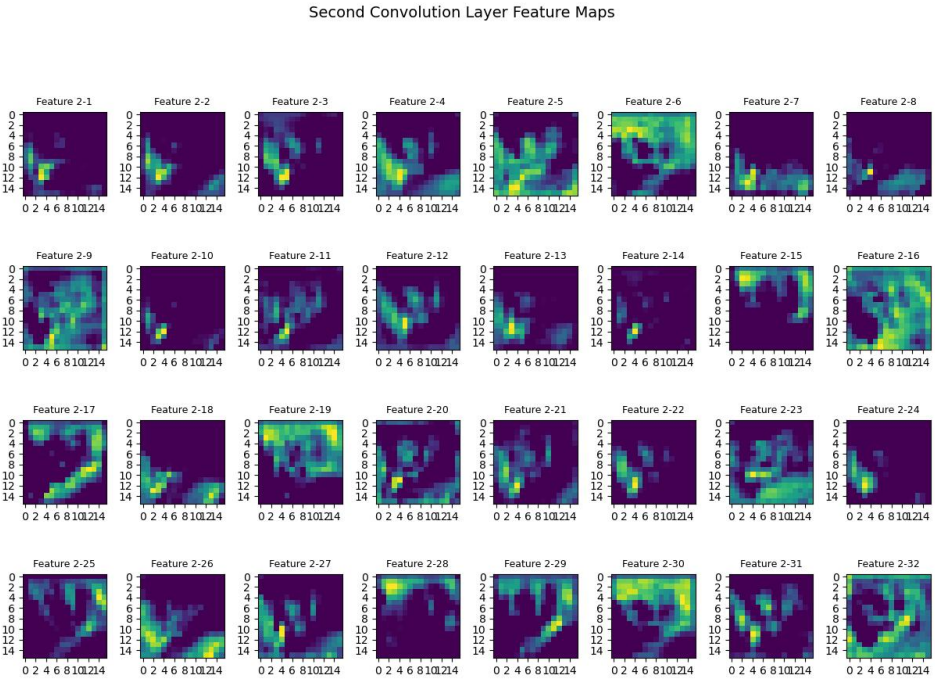

Figure 13. The characteristic diagram of the second convolution layer.

By conducting a thorough analysis of the pooling convolution kernels, feature maps, and image classification prediction results, we discovered that the pooling layer based on the Choquet integral, through the incorporation of fuzzy integral theory, is capable of more effectively capturing and aggregating critical information within images. This leads to an enhancement in classification performance. The proposed method maintains high classification accuracy on the dataset, particularly excelling when accounting for inter-feature relationships in images. Furthermore, it exhibits superior robustness and generalization capabilities. Consequently, the introduction of the Choquet integral-based pooling layer represents a significant and necessary advancement.

Image recognition refers to the process of detecting, identifying, and locating the target objects in an image. It typically encompasses multiple research directions such as object detection, image classification, and semantic segmentation, aiming to comprehensively understand the content and structural information of the image. Image classification is a fundamental task that involves determining the category of an entire image. Its core objective is to categorize the input image into the predefined categories, and it is an important foundational branch in the field of image recognition. In the following, we mainly study the application of three pooling methods in the image classification task and the analysis of their accuracy rates.

4.3. Comparison analysis of pooling methods based on prediction results

CIFAR-10 is a highly representative public dataset in the field of computer vision, mainly used for the development and evaluation of algorithms for image classification tasks. It serves as an important benchmark dataset for beginners and researchers in machine learning and deep learning. The dataset covers 10 different object categories, namely: plane, car, bird, cat, deer, dog, frog, horse, ship, and truck. Each category contains 6000 images, and the same category has various forms, angles, and lighting conditions. For instance, the "ship" category includes ships in different angles and states, eliminating the possibility of chance. On the other hand, some images have backgrounds with similar features to the main objects, increasing the difficulty of classification.

In order to illustrate the validity of the Choquet integral, we conducted the verification on the CIFAR-10 dataset and compared the prediction results of the three pooling methods. First, the prediction results of the neural network that has been trained for 10 cycles and which randomly selected 9 images were compared. Figure 14 shows the prediction result of maximum pooling, Figure 15 shows the prediction result of mean pooling, and Figure 16 shows the prediction result of Choquet integral pooling.

By comparing, it was detected that the three pooling methods produced different prediction results for the same image. For example, for the first image, the prediction result of the maximum pooling method is `"car"`, while the prediction results of the mean pooling method and the Choquet integral pooling method are `"ship"`. It can be clearly seen from the figure that the true result of the first image is `"ship"`. In addition, in the fourth image, the prediction results of the maximum pooling method and the Choquet integral pooling method are both `"frog"`, but the prediction result of the mean pooling method is `"dog"`. From the image, it can be seen that the true result of this image is `"frog"`. Therefore, with respect to images of varying perspectives and states, the predicted results of the three pooling methods are different from the real results. However, it is worth mentioning that for Choquet integral pooling, it not only considers the relationship between objects, but also considers the relationship between features, as well as the relationship between the angle, pose, and layout. So it can be seen from the figure that among 9 randomly selected images from the CIFAR-10 dataset after 10 cycles of

training, the prediction results of Choquet integral pooling are better than those of traditional pooling. This further illustrates that it makes sense to propose Choquet integral pooling.

To mitigate stochasticity and illustrate the stability of the Choquet integral pooling method, the neural network is trained for 20 cycles, and 12 images are randomly selected to compare and analyze the prediction results of the three pooling methods. Figure 17 shows the prediction result of maximum pooling, Figure 18 shows the prediction result of mean pooling, and Figure 19 shows the prediction result of Choquet integral pooling.

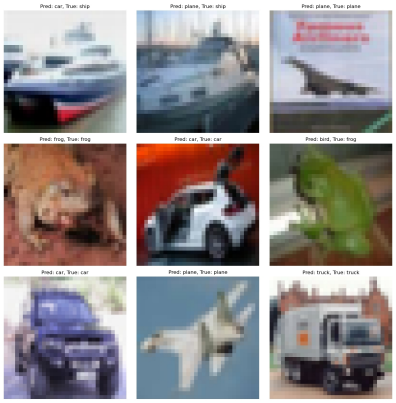

Figure 14. The prediction result of max pooling.

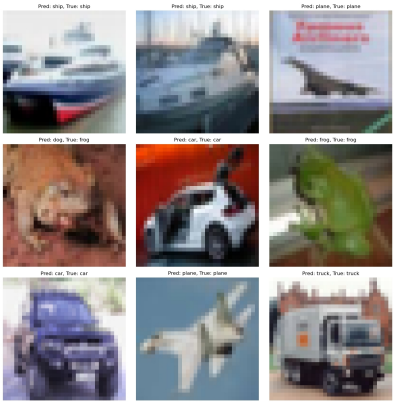

Figure 15. The prediction result of mean pooling.

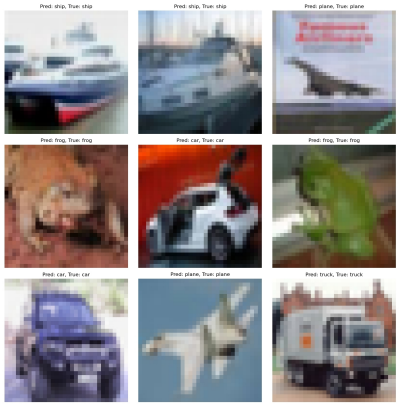

Figure 16. The prediction result of Choquet integral pooling.

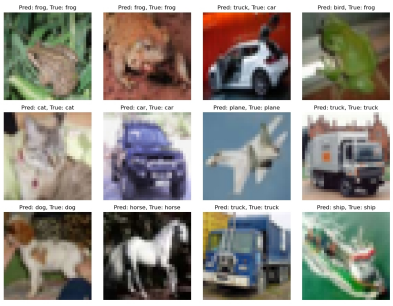

Figure 17. The prediction result of max pooling.

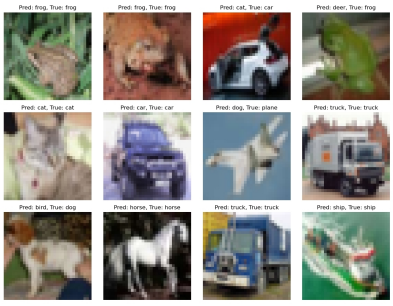

Figure 18. The prediction result of mean pooling.

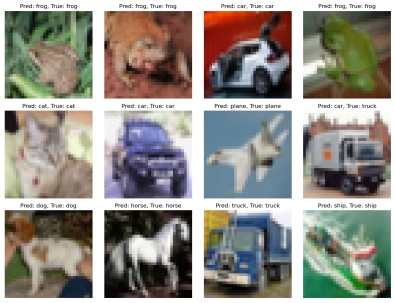

Figure 19. The prediction result of Choquet integral pooling.

Through the application of the aforementioned three pooling methods, the max pooling method produced prediction results that differed from the actual results in 2 out of 12 images. The predictions of the mean pooling method deviated from the real results in 4 out of 12 images. Notably, only one prediction result differed from the actual results when utilizing the newly proposed Choquet integral pooling method. These findings demonstrate that the Choquet integral pooling method outperforms traditional pooling methods when spatial dependence is considered during the image recognition process.

To demonstrate the rationality and effectiveness of the Choquet integral pooling method, a comparison is made of the prediction results obtained from three pooling methods after varying periods of training. Specifically, the results of predicting 16 images across training cycles ranging from 10 to 20, as well as the results of predicting 36 images, are randomly selected for comparative analysis.

As illustrated in Table 4, as the number of cycles increases at 16 pictures predicted by all three methods, Choquet integral pooling demonstrates performance comparable to max pooling, thereby validating the rationality and interpretability of this model. While mean pooling exhibits superior performance during the early stages of training, it tends to be less stable and is more susceptible to overfitting in later stages. In contrast, Choquet integral pooling exhibits more robust generalization capabilities and a stronger resistance to overfitting.

As illustrated in Tables 4 and 5, as the number of cycles increases for the prediction of 36 images using all three methods, Choquet integral pooling and mean pooling yield comparable results. This suggests that Choquet integral pooling offers greater interpretability. In contrast, with an increasing number of predicted images, max pooling exhibits instability and a tendency to overfit in later stages. However, Choquet integral pooling consistently demonstrates robust stability and anti-overfitting capabilities. These findings clearly indicate that Choquet integral pooling possesses significant advantages over traditional pooling methods in image classification tasks.

By comparing and analyzing the three pooling methods from the aspects of convolution kernels, feature maps, and prediction results, it is discovered that max pooling captures the most important features in the image in the process of image recognition and classification, and ignores other features. Mean pooling captures the equal contribution of features in the image without considering the structural dependency between features. Choquet integral pooling makes up for this gap. It weights and aggregates the spatial structural dependency between features through the fuzzy measure, and finally achieves better results. Moreover, since different images exhibit varying degrees of blur, it is essential to introduce Choquet integral pooling for images whose classification relies more on spatial relationships.

Table 4. Analysis of prediction results in different periods.

	Max pooling	Mean pooling	Choquet integral pooling
10 cycles of 16 images	$\frac{13}{16}$	$\frac{15}{16}$	$\frac{13}{16}$
20 cycles of 16 images	$\frac{13}{16}$	$\frac{13}{16}$	$\frac{13}{16}$
Rate of change	0	$-\frac{1}{8}$	0

Table 5. Analysis of prediction results in different periods.

	Max pooling	Mean pooling	Choquet integral pooling
10 cycles of 36 images	$\frac{33}{36}$	$\frac{25}{36}$	$\frac{22}{36}$
20 cycles of 36 images	$\frac{29}{36}$	$\frac{29}{36}$	$\frac{26}{36}$
Rate of change	$-\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$

5. Conclusions and prospects

In this paper, we first studied the properties of the C_C-integral, which is a generalization of the Choquet integral. In the process of research, it was found that the properties satisfied by the Choquet integrals are not necessarily satisfied by the C_C-integral. While considering the properties of Choquet integrals, the properties of Copula were also taken into account, where the copula can be a triangular norm or an overlap function. Second, when the Copula function is $C(x, y) = xy$, the C_C-integral is a standard Choquet integral. It is an aggregation function that effectively handles the interaction between features, thereby compensating for the limitations of traditional pooling methods in processing such image features. Thus we proposed a new pooling method, namely, Choquet integral pooling, which applies the Choquet integral in the pooling layer for feature extraction, and validated this approach on the CIFAR-10 dataset. A comparative analysis was conducted from three perspectives: convolution kernels, feature maps, and the classification results of images. The findings demonstrated that the proposed Choquet integral pooling outperformed traditional pooling methods when dealing with images characterized by indistinct features or requiring consideration of interdependencies among features. This is attributed to the use of the fuzzy measure in the Choquet integral calculation process, which effectively weights and aggregates feature relationships to enhance performance. Additionally, as the number of training epochs increased, the Choquet integral pooling method exhibited a favorable growth trend. Within the same number of training epochs, it demonstrated superior stability and anti-overfitting capabilities compared to conventional pooling techniques. Consequently, the proposed Choquet integral pooling not only achieves a unique balance between global and local features, thereby more precisely capturing critical information in feature maps and improving the efficiency of pooling operations, but also exhibits enhanced stability and robustness.

This paper only attempts to apply the Choquet integral to CNN. The application of copula functions, particularly when they are set as overlap functions or other types of functions in a CNN, remains an area yet to explored. On the other hand, whether other generalizations of the Choquet integral, such as the C_F-integral [16] or the d-Choquet integral [45], can be applied in a CNN needs further study.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this articl.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 12161082) and Gansu Province Outstanding Youth Fund project (Grant No. 24JRRA121). The authors are very grateful to the anonymous referees for their valuable suggestions.

Conflict of interest

The authors declare that they have no conflict of interest.

Author contributions

Conceptualization, H. Y.; Methodology, J. X. W.; Formal analysis, H. Y. and J. X. W.; Writing—original draft preparation, J. X. W; Supervision, H. Y. All authors have read and agreed to the published version of the manuscript.

References

1. G. Beliakov, H. Bustince, T. Calvo, *A Practical Guide to Averaging Functions*, Springer, Cham, Switzerland, 2016. https://doi.org/10.1007/978-3-319-24753-3
2. E. Barrenechea, H. Bustince, J. Fernandez, D. Paternain, J. A. Sanz, Using the Choquet integral in the fuzzy reasoning method of fuzzy rule-based classification systems, *Axioms*, **2** (2013), 208–223. https://doi.org/10.3390/axioms2020208
3. D. Paternain, J. Fernandez, H. Bustince, R. Mesiar, G. Beliakov, Construction of image reduction operators using averaging aggregation functions, *Fuzzy Sets Syst.*, **261** (2015), 87–111. https://doi.org/10.1016/j.fss.2014.03.008
4. L. M. Rodrigues, G. P. Dimuro, D. T. Franco, J. C. Fachinello, A system based on interval fuzzy approach to predict the appearance of pests in agriculture, in *2013 Joint IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS)*, Edmonton, AB, Canada, 2003, 1262–1267. https://doi.org/10.1109/IFSA-NAFIPS.2013.6608583
5. C. Campomanes-Alvarez, O. Ibanez, O. Cordon, C. Wilkinson, Hierarchical information fusion for decision making in craniofacial superimposition, *Inf. Fusion*, **39** (2018), 25–40. https://doi.org/10.1016/j.inffus.2017.03.004
6. Q. Zhang, L. Yang, Z. Chen, P. Li, A survey on deep learning for big data, *Inf. Fusion*, **42** (2018), 146–157. https://doi.org/10.1016/j.inffus.2017.10.006
7. G. Choquet, Theory of capacities, *Ann. de Inst. Fourier*, **5** (1954), 131–295. https://doi.org/10.5802/AIF.53
8. J. C. R. Alcantud, The capacity compliance problem: Refinements of time discounting to Choquet integrals with 2-additive fuzzy measures, *Comput. Appl. Math.*, **44** (2025), 424. https://doi.org/10.1007/s40314-025-03407-4
9. M. Skublewska-Paszkowska, P. Karczmarek, P. Powroznik, E. Lukasik, J. Smolka, M. Dolecki, Aggregation of tennis multivariate time-series using the Choquet integral and its generalizations, in *2023 IEEE International Conference on Fuzzy Systems (FUZZ)*, Incheon, Korea, Republic of, 2023, 1–6. https://doi.org/10.1109/FUZZ52849.2023.10309746
10. J. J. Huang, C. Y. Chen, Temporally adaptive hierarchical Choquet integrals: A measure-theoretic framework for dynamic non-additive integration in approximate reasoning, *Fuzzy Sets Syst.*, **520** (2025), 109550. https://doi.org/10.1016/j.fss.2025.109550
11. M. Sugeno, T. Murofushi, Pseudo-additive measure and integral, *J. Math. Anal. Appl.*, **122** (1987), 197–222. https://doi.org/10.1016/0022-247X(87)90354-4

12. M. Grabisch, Fuzzy measure and integrals: Recent developments, in: D. Tamir, N. Rishe, A. Kandel (eds), in *Fifty Years of Fuzzy Logic and its Applications*, Springer International, Cham, Switzerland, 2015, 9–22. https://doi.org/10.1007/978-3-319-19683-1_8

13. R. Mesiar, Choquet-like integral, *J. Math. Anal. Appl.*, **194** (1995), 477–488. https://doi.org/10.1006/jmaa.1995.1312

14. G. Lucca, J. A. Sanz, G. P. Dimuro, B. Bedregal, R. Mesiar, A. KolesÃ¡rovÃ¡, et al., Pre-aggregation functions: Construction and an application, *IEEE Trans. Fuzzy Syst.*, **24** (2016), 260–272. https://doi.org/10.1109/TFUZZ.2015.2453020

15. G. Lucca, J. A. Sanz, G. P. Domuro, B. Bedregal, M. J. Asiain, M. Eikano, et al., Cc-Integral: Choquet-like copula-based aggregation functions and its application in fuzzy rule-based classification system, *Knowl.-Based Syst.*, **119** (2017), 32–43. https://doi.org/10.1016/j.knosys.2016.12.004

16. G. Lucca, J. A. Sanz, G. P. Dimuro, B. Bedregal, H. Bustince, R. Mesiar, CF-integrals: A new family of pre-aggregation functions with application to fuzzy rule-based classification systems, *Inf. Sci.*, **435** (2018), 94–110. https://doi.org/10.1016/j.ins.2017.12.029

17. P. G. Dimuro, J. Fernandez, B. Bedregal, R. Mesiar, J. A. Sanz, G. Lucca, et al., The state-of-art of the generalization of the Choquet integral: From aggregation and pre-aggregation to ordered directionally monotone function, *Inf. Fusion*, **57** (2020), 27–43. https://doi.org/10.1016/j.inffus.2019.10.005

18. A. Krizhevsky, I. Sutskever, G. E. Hinton, ImageNet classification with deep convolutional neural networks, *Commun. ACM*, **60** (2017), 84–90. https://doi.org/10.1145/3065386

19. R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, 2014, 580–587. https://doi.org/10.1109/CVPR.2014.81

20. R. Collobert, J. Weston, A unified architecture for natural language processing: Deep neural networks with multitask learning, in *ICML '08: Proceedings of the 25th International Conference on Machine Learning*, 2008, 160–167. https://doi.org/10.1145/1390156.1390177

21. M. Lin, Q. Chen, S. Yan, Network In Network, in *International Conference on Learning Representations*, 2014. Available from: https://openreview.net/forum?id=ylE6yojDR5yqX.

22. D. Yu, H. Wang, P. Chen, Z. Wei, Mixed pooling for convolutional neural networks, in *Rough Sets and Knowledge Technology*, **8818** (2014), 364–375. https://doi.org/10.1007/978-3-319-11740-9_34

23. K. He, X. Zhang, S. Ren, J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, *IEEE Trans. Pattern Anal. Mach. Intell.*, **37** (2015), 1904–1916. https://doi.org/10.1109/TPAMI.2015.2389824

24. B. Graham, Fractional max-pooling, preprint, arXiv:1412.6071.

25. S. Sabour, N. Frosst, G. E. Hinton, Dynamic routing between capsules, in *NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2017, 3859–3869. Available from: https://dlnext.acm.org/doi/10.5555/3294996.3295142.

26. C. Y. Lee, P. Gallagher, Z. Tu, Generalizing pooling functions in convolutional neural networks: Mixed, gated, and tree, *IEEE Trans. Pattern Anal. Mach. Intell.*, **40** (2018), 863–875. https://doi.org/10.1109/TPAMI.2017.2703082

27. I. Rodriguez-Martinez, J. Lafuente, R. H. N. Santiago, G. P. Dimuro, F. P. Herrera, H. Bustince, Replacing pooling functions in Convolutional Neural Networks by linear combinations of increasing functions, *Neural Networks*, **152** (2022), 380–393. https://doi.org/10.1016/j.neunet.2022.04.028

28. I. Rodriguez-Martinez, T. da Cruz Asmus, G. P. Dimuro, F. Herrera, Z. Takáč, H. Bustince, Generalizing max pooling via (a,b)-grouping functions for convolutional neural networks, *Inf. Fusion*, **99** (2023), 101893. https://doi.org/10.1016/j.inffus.2023.101893

29. M. Ferrero-Jaurrieta, R. Paiva, A. Cruz, B. R. Callejas Bedregal, L. De Miguel, Z. Takáč, et al., Non-symmetric over-time pooling using pseudo-grouping functions for convolutional neural networks, *Eng. Appl. Artif. Intell.*, **133** (2024), 108470. https://doi.org/10.1016/j.engappai.2024.108470

30. C. Dias, J. Bueno, E. Borges, G. Lucca, H. Santos, G. Dimuro, et al., Simulating the behaviour of Choquet-like (pre) aggregation gunctions for image resizing in the pooling layer of deep learning networks, in *Fuzzy Techniques: Theory and Applications*, Springer, Cham, 2019, 224–236. https://doi.org/10.1007/978-3-030-21920-8_21

31. C. A. Dias, J. C. S. Bueno, E. N. Borges, S. S. C. Botelho, G. P. Dimuro, G. Lucca, et al., Using the Choquet integral in the pooling layer in deep learning, in *Fuzzy Information Processing*, Springer, Cham, 2018, 144–154. https://doi.org/10.1007/978-3-030-03023-0_14

32. G. Beliakov, A. Pradera, T. Calvo, *Aggregation Functions: A Guide for Practitioners*, Springer Berlin, Heidelberg, 2007. https://doi.org/10.1007/978-3-540-73721-6

33. E. P. Klement, R. Mesiar, E. Pap, *Triangular Norms*, Springer Dordrecht, 2000. https://doi.org/10.1007/978-94-015-9540-7

34. G. Mayor, E. Trillas, On the representation of some aggregation functions, in *Proceedings of the 16th IEEE International Symposium on Multiple-Valued Logic (ISMVL)*, Los Alamitos, 1986, 110–114.

35. C. Alsina, M. J. Frank, B. Schweizer, *Associative Functions. Triangular Norms and Copulas*, World Scientific, 2006, 1–360. https://doi.org/10.1142/9789812774200

36. B. Schweizer, A. Sklar, *Probabilistic Metric Spaces*, North-Holland, New York, 1983.

37. D. Gomez, J. T. Rodriguez, J. Montero, H. Bustince, E. Barrenechea, n-dimensional overlap functions, *Fuzzy Sets Syst.*, **287** (2016), 57–75. https://doi.org/10.1016/j.fss.2014.11.023

38. M. Elkano, M. Galar, J. Sanz, H. Bustince, Fuzzy rule-based classification systems for multi-class problems using binary decomposition strategies: On the influence of n-dimensional overlap functions in the fuzzy reasoning method, *Inf. Sci.*, **332** (2016), 94–114. https://doi.org/10.1016/j.ins.2015.11.006

39. M. Elkano, M. Galar, J. A. Sanz, A. Fernandez, E. Barrenechea, F. Herrera, et al., Enhancing multi-class classification in FARC-HD fuzzy classifier: on the synergy between n-dimensional overlap functions and decomposition strategies, *IEEE Trans. Fuzzy Syst.*, **23** (2015), 1562–1576. https://doi.org/10.1109/TFUZZ.2014.2370677

40. H. Bustince, J. Fernandez, A. Kolasarova, R. Mesiar, Directional monotonicity of fusion, *Eur. J. Oper. Res.*, **244** (2015), 300–308. https://doi.org/10.1016/j.ejor.2015.01.018

41. T. Wilkin, G. Beliakov, Weakly monotonic averaging functions, *Int. J. Intell. Syst.*, **30** (2015), 144–169. https://doi.org/10.1002/int.21692

42. T. Murofushi, M. Sugeno, M. Machida, Non-monotonic fuzzy measures and the Choquet integral, *Fuzzy Sets Syst.*, **64** (1994), 73–84. https://doi.org/10.1016/0165-0114(94)90008-6

43. H. Yang, L. Shang, Z. Gong, C_C-integral on interval-valued Sugeno probability measure and its application in multi-criteria decision-making rroblems, *Axioms*, **11** (2022), 317. https://doi.org/10.3390/axioms11070317

44. J. Qiao, B. Q. Hu, The distributive laws of fuzzy implications over overlap and grouping functions, *Inf. Sci.*, **438** (2018), 107–126. https://doi.org/10.1016/j.ins.2018.01.047

45. H. Bustince, R. Mesiar, J. Fernandez, M. Galar, D. Paternain, A. Altalhi, et al. d-Choquet integrals: Choquet integrals based on dissimilarities, *Fuzzy Sets Syst.*, **414** (2021), 1–27. https://doi.org/10.1016/j.fss.2020.03.019

AIMS Press