



Evaluación de estrategias de binarización en la clasificación de imágenes usando deep learning

Francisco Pérez, Siham Tabik, Alberto Castillo, Hamido Fujita* y Francisco Herrera
Departamento de Ciencias de la Computación e Inteligencia Artificial, Universidad de Granada,

Granada, España

{fperezhernandez,siham}@ugr.es,{albertocl,herrera}@decsai.ugr.es

*Iwate Prefectural University, Takizawa, Iwate, Japan

Resumen—El reconocimiento de objetos pequeños en imágenes con redes neuronales convolucionales (CNNs) sigue siendo un reto, especialmente cuando estos objetos se manipulan con la mano de forma muy similar. En este trabajo proponemos dividir un problema multiclase del ámbito de la seguridad en un problema de binarización de clases para obtener un mejor resultado. Nuestro objetivo ha sido obtener el mejor modelo de aprendizaje que distinga entre 6 clases, pistolas, smartphone, billete, monedero, tarjeta y background. Concretamente, evaluamos las técnicas *One-Versus-All* (OVA), *One-Versus-One* (OVO) y *Distance-based Relative Competence Weighting combination para OVO* (DRCW-OVO) basadas en CNNs. El mejor rendimiento se obtiene usando DRCW-OVO con una precisión de 90,47 %, un recall del 90,93 % y un F1 del 90,59 %. Esto significó una mejora del 2,58 % en precisión, 1,48 % en recall y 2,13 % en F1 frente a un multclasificador normal.

Index Terms—Clasificación, Convolutional Neuronal Networks (CNNs), Multclasificación, Deep Learning, Machine Learning, *One-Versus-All* (OVA), *One-Versus-One* (OVO), DRCW-OVO, ResNet-101

I. INTRODUCCIÓN

La tarea de clasificar imágenes es un reto en nuestros días y se puede ver en la rama de visión por computador. En competiciones como ImageNet [1], cada año participan muchos equipos de diferentes grandes empresas como Google, y la diferencia para obtener los mejores resultados es crucial.

En el paradigma de *machine learning* clásico, la clasificación es un problema bien conocido donde implementar nuevas técnicas para mejorar los resultados. Es el caso de la descomposición de un problema multiclase en problemas biclase. La clásica técnica *One-Versus-All* (OVA) y *One-Versus-One* (OVO) se usa en muchos trabajos como un buen instrumento para aumentar el rendimiento de los modelos.

ImageNet y COCO (*Common Objects in Context*) [2], abordan un problema de clasificación de imágenes. Concretamente, la tarea es diferenciar objetos cotidianos que pueden provocar una confusión entre ellos. Un problema bien conocido por nuestro grupo de investigación es la detección de armas de fuego. Este tipo de imágenes suponen un reto ya que hay muchos objetos que se pueden manejar de la misma forma.

La mayoría de trabajos anteriores en este ámbito abordaban la detección de armas en imágenes de rayos X, milimétricas o RGB utilizando métodos clásicos de *machine learning* que requieren una alta intervención humana [3], [4], [5], [6], [7].

Actualmente, los modelos más precisos en la clasificación de imágenes y detección de objetos se basan en redes neuronales convolucionales profundas (CNNs) [8]. Estos modelos aprenden automáticamente las características distintivas de los objetos a partir de un gran conjunto de datos etiquetados.

Por lo que sabemos, el primer modelo de detección automática de pistolas en vídeo basado en CNNs fue desarrollado por Olmos et al en [9]. Sin embargo, cuando en los vídeos, una persona manipula objetos como smartphone, billete, monedero o tarjeta, el modelo produce falsos positivos. La Figura 1 muestra ejemplos de este tipo de falsos positivos cometidos por el modelo de detección. Esto puede explicarse por el hecho de que el modelo aprendió la forma en que se manejan las pistolas, siendo también una característica clave, lo cual es intolerable en el campo de la videovigilancia debido a todas las posibles falsas alarmas que se puedan producir.

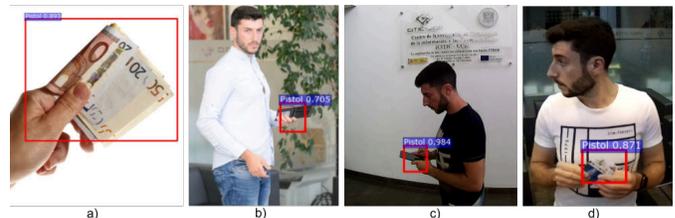


Figura 1. Falsos positivos cometidos por el modelo de detección, donde se confunde con a) billete, b) monedero, c) smartphone y d) tarjeta.

Para mejorar la precisión y robustez del modelo, (1) construimos un conjunto de datos de entrenamiento de alta calidad que incluye todas las clases de objetos posibles que se manejan comúnmente de manera similar, (2) desarrollamos un modelo de clasificación robusto y (3) mejoramos la robustez usando técnicas de *machine learning* como OVA y OVO [10]. Además, el método *Distance-based Relative Competence Weighting combination para OVO* (DRCW-OVO) [11], para problemas multiclase, es utilizado como una extensión de los métodos clásicos de agregación de OVO. Por lo tanto, nos centramos en la tarea de clasificación, ya que es la base de la detección. De hecho, los modelos de detección más influyentes combinan un modelo de clasificación con un método de búsqueda de regiones [12], [13], [14].

Este trabajo se organiza analizando, en la Sección II, los preliminares, donde se analizan trabajos relacionados y las

estrategias de descomposición en problemas multiclase. En la sección III la construcción de la base de datos. La evaluación de las estrategias usadas se encuentra en la sección IV y finalmente las conclusiones aparecen en la sección V.

II. PRELIMINARES

La mayoría de los trabajos analizados hacen uso de OVA y OVO en tareas visuales, reconocimiento de objetos, clasificación de imágenes y segmentación de imágenes, utilizando sólo modelos clásicos como *Support Vector Machine* (SVM), *Linear Discriminant Analysis* (LDA) y *k-Nearest Neighbors* (k-NN). Por ejemplo, en clasificación de imágenes, los autores en [15] analizaron el enfoque OVA y OVO para reducir el espacio de las características en tres dataset bien conocidos, MNIST, *Amsterdam Library of Object Images* (ALOI) y *Australian Sign Language* (Auslan). Para la estimación de la pose en la segmentación de imágenes, los autores en [16] compararon un clasificador individual basado en CNN con OVA y OVO basado en SVM y mostraron que las CNN logran un rendimiento ligeramente mejor que OVA y OVO basados en SVM. De manera similar, en la tarea de clasificación de imágenes *remote sensing*, los autores en [17] también compararon OVA y OVO basados en SVM y 1-NN y concluyeron que OVA proporcionó peores resultados debido al desequilibrio entre clases. Los mejores resultados fueron obtenidos por OVO con SVM. En el reconocimiento facial, los autores de [18] utilizaron un modelo basado en CNN para la extracción de características y un SVM, OVA y OVO para la clasificación. Los mejores resultados fueron obtenidos por CNN en combinación con SVM. Los autores en [19] compararon la técnica *Half-Against-Half* (HAH) con OVA y OVO en la clasificación de imágenes y encontraron que HAH proporciona resultados similares o peores en los puntos de referencia evaluados. Nuestro trabajo se diferencia de todos los anteriores en que mejora la robustez del reconocimiento de objetos que se manejan similarmente utilizando OVA, OVO y DRCW-OVO basados en CNNs. Hasta donde sabemos, ningún trabajo previo aplicó estas técnicas en modelos de Deep Learning para la clasificación de imágenes.

Los problemas de clasificación que involucran múltiples clases son más difíciles de resolver. El enfoque común para abordar este tipo de problemas es reformular el problema original multiclase en un conjunto de problemas binarios de dos clases. Las técnicas más comunes en este contexto son OVA [20] y OVO [21]. En [22] se ofrece una explicación ampliada y completa de todos los posibles métodos de agregación para OVA y OVO. Además, el método DRCW-OVO [11] amplía el clásico OVO utilizando la distancia entre clases.

II-A. OVA

La estrategia *One-Versus-All* (OVA) reformula el problema de la clasificación multiclase en un conjunto de clasificadores binarios donde cada clasificador aprende cómo distinguir cada clase individual contra el resto de clases juntas. Este enfoque produce tantos clasificadores como el número de clases en el problema original. La predicción final se calcula combinando las predicciones de los clasificadores individuales mediante

un método de agregación denominado *Maximum Confidence Strategy* (MAX). La clase con el mayor número de votos se considera como la clase pronosticada. Formalmente,

$$\text{PredictedClass} = \arg \max_{i=1, \dots, m} r_i$$

, donde $r_i \in [0, 1]$ es la confianza para la clase i y m es el número de clases. En la Figura 2 se ilustra OVA aplicado al problema multiclase considerado en este trabajo.

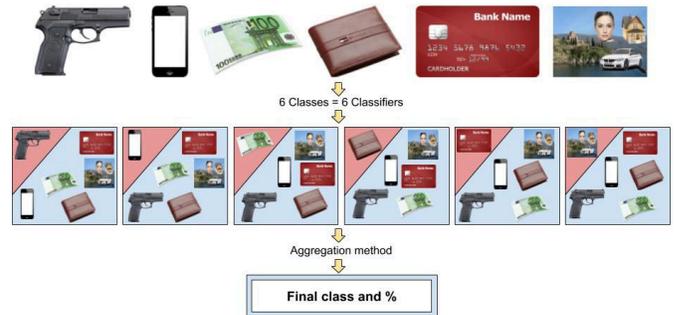


Figura 2. Proceso OVA.

II-B. OVO

La estrategia *One-Versus-One* (OVO) reformula el problema multiclase original en tantos problemas binarios como combinaciones posibles entre pares de clases para que cada clasificador aprenda a discriminar entre cada par. Es decir, un problema m -clases se convertirá en $m(m-1)/2$ clasificadores y en el caso considerado, con $m = 6$ clases, se reformulará en 15 clasificadores, como se aprecia en la Figura 3.



Figura 3. Proceso OVO.

El sistema OVO puede utilizar diversas estrategias de agregación como pueden ser *Max-Wins* (VOTE aleatorio o por peso), *Weighted Voting Strategy* (WV), *Learning Valued Preference for Classification* (LVPC), *Preference Relations Solved by Non-Dominance Criterion* (ND), *Classification by Pairwise Coupling* (PC) y *Wu, Lin and Weng Probability Estimates by Pairwise Coupling approach* (PE).

VOTE random en OVO: La regla VOTE, también llamada regla *Max-Wins* [23], se considera como la regla básica de decisión en OVO. En este método se repasa la matriz y en el elemento r_{ij} si la posibilidad de pertenecer a la clase i es superior a 0,5, el resultado es la clase i . Finalmente, se repasa



toda la matriz, se suma el resultado y se selecciona la clase con más votos. Si tenemos dos o más clases con el mismo número de votos, seleccionamos una al azar. Formalmente,

$$\text{PredictedClass} = \arg \max_{i=1, \dots, m} \sum_{i \leq j \neq i \leq m} s_{ij}$$

donde s_{ij} es 1 si $r_{ij} > r_{ji}$ y 0 en caso contrario.

VOTE weight en OVO: Este es un nuevo enfoque propuesto. En el método VOTE, podemos tener 2 o más clases con los mismos votos. Para las clases con mayor número de votos, proponemos sumar las predicciones y seleccionar la clase con el valor máximo como la clase final.

WV en OVO: La técnica *Weighted Voting strategy* pretende obtener la clase con la mayor probabilidad. Por esta razón, cada clase suma sus predicciones y la clase con el valor máximo es el resultado final. La regla de decisión es:

$$\text{PredictedClass} = \arg \max_{i=1, \dots, m} \sum_{i \leq j \neq i \leq m} r_{ij}$$

LVPC en OVO: Learning Valued Preference for Classification (LVPC) [24], [25] es la técnica que utiliza el peso de las clases y que penaliza a las clases que no tienen suficiente certeza. La regla de decisión es:

$$P_{ij} = r_{ij} - \min\{r_{ij}, r_{ji}\}; P_{ji} = r_{ji} - \min\{r_{ij}, r_{ji}\}$$

$$C_{ij} = \min\{r_{ij}, r_{ji}\}; I_{ij} = 1 - \max\{r_{ij}, r_{ji}\}$$

$$\text{Class} = \arg \max_{i=1, \dots, m} \sum_{i \leq j \neq i \leq m} P_{ij} + \frac{1}{2} C_{ij} + \frac{N_i}{N_i + N_j} I_{ij}$$

donde N_i es el número de ejemplos de la clase i en train.

ND en OVO: La técnica *Preference Relations Solved by Non-Dominance Criterion* (ND) se definió originalmente para la toma de decisiones con relaciones de preferencia difusas [26]. En [27] se aplica el mismo criterio en un sistema de clasificación OVO. Primero normalizando, seguido de calcular la preferencia difusa y calcular el grado para cada clase, para obtener la clase final:

$$\bar{r}_{ij} = \frac{r_{ij}}{r_{ij} + r_{ji}}$$

$$r'_{ij} = \begin{cases} \bar{r}_{ij} - \bar{r}_{ji}, & \text{cuando } \bar{r}_{ij} > \bar{r}_{ji} \\ 0, & \text{de otra manera} \end{cases}$$

$$ND_i = 1 - \sup_{j \in C} [r'_{ji}]$$

$$\text{Class} = \arg \max_{i=1, \dots, m} ND_i$$

PC en OVO: La técnica *Classification by Pairwise Coupling* (PC) [28] intenta mejorar la estrategia de votación cuando los resultados de los clasificadores son probabilidades estimadas de clase. Este método estima la probabilidad conjunta para todas las clases a partir de las probabilidades de clase por pares de los clasificadores binarios. El algoritmo es:

1. Inicialización:

$$\hat{p}_i = \frac{2 \sum_{1 \leq j \neq i \leq m} r_{ij}}{m(m-1)} \text{ para todo } i = 1, \dots, m$$

$$\hat{\mu}_{ij} = \frac{\hat{p}_i}{\hat{p}_i + \hat{p}_j} \text{ para todo } i, j = 1, \dots, m$$

2. Repetir hasta converger:

$$\hat{p}_i = \hat{p}_i \frac{\sum_{1 \leq j \neq i \leq m} n_{ij} r_{ij}}{\sum_{1 \leq j \neq i \leq m} n_{ij} \hat{\mu}_{ij}} \text{ para todo } i = 1, \dots, m$$

donde n_{ij} es el número de elementos en train en las clases i th y j th.

$$\hat{p}_i = \frac{\hat{p}_i}{\sum_{i=1}^m \hat{p}_i} \text{ para todo } i = 1, \dots, m$$

$$\hat{\mu}_{ij} = \frac{\hat{p}_i}{\hat{p}_i + \hat{p}_j} \text{ para todo } i, j = 1, \dots, m$$

Finalmente, la salida de la clase será:

$$\text{Class} = \arg \max_{i=1, \dots, m} \hat{p}_i$$

PE en OVO: La técnica *Wu, Lin y Weng Probability Estimates by Pairwise Coupling Approach* (PE) [29] es similar a PC, estima las probabilidades (p) de cada clase a partir de las probabilidades por pares. PE optimiza el siguiente problema:

$$\min_p \sum_{i=1}^m \sum_{1 \leq j \neq i \leq m} (r_{ji} p_i - r_{ij} p_j)^2 \text{ sujeto a } \sum_{i=1}^k p_i = 1, p_i \geq 0, \forall i$$

II-C. DRCW-OVO

Distance-based Relative Competence Weighting combination para *One-Versus-One* (DRCW-OVO) en problemas multiclase [11] es una extensión de la técnica OVO que pretende mejorar el problema del desbalanceo de las clases usando la distancia con los k elementos vecinos a la nueva instancia.

DRCW-OVO, una vez que se ha obtenido la matriz de pesos:

1. Calcular la distancia media de los k vecinos cercanos a cada clase en el vector \mathbf{d} .
2. Calcular la nueva matriz de pesos R^w de la forma:

$$r_{ij}^w = r_{ij} \cdot w_{ij} \text{ donde } w_{ij} = \frac{d_j^2}{d_i^2 + d_j^2}$$

$$r'_{ij} = \begin{cases} \bar{r}_{ij} - \bar{r}_{ji}, & \text{cuando } \bar{r}_{ij} > \bar{r}_{ji} \\ 0, & \text{de otra manera} \end{cases}$$

siendo d_i la distancia de la instancia a los vecinos cercanos de la clase i .

3. Usar la estrategia *Weighted Voting* (WV) en la nueva matriz de pesos R^w para obtener la clase final.

Se ha calculado, para la distancia entre imágenes, la distancia *Quadratic-Chi* [30] con el histograma de las imágenes:

$$X^2(P, Q) = \frac{1}{2} \sum_i \frac{(P_i - Q_i)^2}{(P_i + Q_i)}$$

donde P_i es el histograma de la nueva instancia y Q_i es la media del histograma de los k vecinos cercanos.

III. CONSTRUCCIÓN DE LA BASE DE DATOS GUIADA POR EL RENDIMIENTO EN CLASIFICACIÓN

El objetivo de esta sección es construir un dataset que permita al modelo de clasificación distinguir entre objetos que se manejan de forma similar. El proceso de construcción se ha guiado por el rendimiento del modelo de clasificación.

Se ha usado el modelo de clasificación ResNet-101 [31] inicializándolo con los pesos preentrenados en ImageNet [1]. Como software de Deep Learning se ha usado Tensorflow [32] y los experimentos se han realizado en una NVIDIA Titan Xp, durando dos horas cada entrenamiento.

Para mejorar el aprendizaje del modelo, se ha desarrollado el conjunto de datos en cinco pasos:

1. Como punto de partida se ha usado el dataset ¹ construido en [9] de imágenes de pistolas, junto a la clase background, donde se han incluido caras, coches, escenas, etc. La mayoría de las imágenes fueron descargadas de Internet.
2. A las imágenes del punto 1, se ha sumado la primera clase competitiva, el smartphone.
3. Se han añadido el resto de objetos que pueden causar confusión como billete, monedero y tarjeta.
4. Enriquecimos todas las clases del conjunto de datos con imágenes tomadas por diferentes cámaras con diversas calidades y resoluciones, una cámara réflex, Nikon D5200, y dos cámaras de videovigilancia, Hikvision DS-2CD2420F-IW y Samsung SNH-V6410PN.
5. Eliminamos las imágenes borrosas junto con las imágenes en las que el ojo humano podía confundir los distintos objetos.

Para evaluar la calidad de cada conjuntos de datos se utilizó la base de datos Dataset-Test. Las características de los conjuntos de datos construidos se proporcionan en la Tabla I.

Tabla I

ELEMENTOS QUE COMPONEN CADA UNO DE LOS 5 DATASET DE TRAIN, Y EL DATASET DE TEST, EN NÚMERO DE IMÁGENES.

Dataset-	# img	Pistola	Smartphone	Billete	Monedero	Tarjeta	Background
1	4616	3464	0	0	0	0	1152
2	5412	3394	866	0	0	0	1152
3	5862	3394	866	134	137	179	1152
4	7177	3523	1022	287	315	307	1723
5	5801	1580	755	545	581	340	2000
Test	1100	294	115	123	104	64	400

Las predicciones, y las medidas precisión, recall y F1 obtenidas en cada clase por el modelo entrenado individualmente en la Dataset-1, -2, -3, -4 y -5 se muestran en la Tabla II. En general, el aumento del número de clases mejora el rendimiento global y por clase del modelo. De la Dataset-5 a Dataset-1, el rendimiento mejoró en 35.7 % en precisión, 8.7 % en recall y 28.05 % en F1. El clasificador reconoce la pistola más apropiadamente cuando aprende a distinguir mejor entre ella y más objetos diferentes como, smartphone, monedero, billete y tarjeta. Como se puede observar, los valores de precisión, recall y F1 obtenidos en las diferentes

¹<http://sci2s.ugr.es/weapons-detection>

Tabla II
PARA CADA DATASET DE TRAIN, PREDICIONES, PRECISIÓN, RECALL Y F1 DE CADA CLASE, TESTEADO EN DATASET-TEST.

Clasificador entrenado con Dataset-1	Background	Billete	Monedero	Pistola	Smartphone	Tarjeta	Precisión	Recall	F1
Predicción	Background	246	58	23	0	25	40	62.91 %	62.19 %
	Pistola	154	65	82	294	90	24	41.46 %	100.00 %
								52.19 %	80.75 %
									MEDIA
Clasificador entrenado con Dataset-2	Background	Billete	Monedero	Pistola	Smartphone	Tarjeta	Precisión	Recall	F1
Predicción	Background	236	40	14	0	1	22	75.39 %	59.00 %
	Pistola	120	40	50	293	21	7	55.17 %	99.65 %
	Smartphone	44	43	40	1	93	35	36.32 %	80.86 %
								55.63 %	79.84 %
									MEDIA
Clasificador entrenado con Dataset-3	Background	Billete	Monedero	Pistola	Smartphone	Tarjeta	Precisión	Recall	F1
Predicción	Background	228	1	6	0	2	3	95.00 %	57.00 %
	Billete	6	108	0	0	0	23	78.83 %	89.80 %
	Monedero	5	1	66	0	1	0	90.41 %	63.46 %
	Pistola	120	4	16	291	15	3	64.81 %	98.97 %
	Smartphone	39	8	15	3	95	6	57.97 %	82.60 %
	Tarjeta	2	1	1	0	2	29	82.85 %	45.41 %
								78.18 %	72.52 %
									MEDIA
Clasificador entrenado con Dataset-4	Background	Billete	Monedero	Pistola	Smartphone	Tarjeta	Precisión	Recall	F1
Predicción	Background	202	5	11	2	6	4	90.60 %	75.50 %
	Billete	0	108	0	0	0	5	95.57 %	87.80 %
	Monedero	8	1	64	1	5	0	81.01 %	61.53 %
	Pistola	70	4	11	291	6	1	75.97 %	98.97 %
	Smartphone	18	3	15	0	97	5	70.28 %	85.96 %
	Tarjeta	2	2	0	0	1	49	90.74 %	76.56 %
								84.04 %	80.78 %
									MEDIA
Clasificador entrenado con Dataset-5	Background	Billete	Monedero	Pistola	Smartphone	Tarjeta	Precisión	Recall	F1
Predicción	Background	338	3	7	1	1	1	96.29 %	84.50 %
	Billete	7	114	1	1	0	4	89.76 %	92.68 %
	Monedero	9	4	91	2	16	0	74.59 %	87.50 %
	Pistola	34	0	15	290	1	1	88.68 %	98.63 %
	Smartphone	8	1	2	0	97	1	88.99 %	84.34 %
	Tarjeta	4	1	2	0	0	57	89.06 %	89.06 %
								87.89 %	89.45 %
									MEDIA

clases al entrenar el modelo en Dataset-1, -2 y -3 son muy desequilibrados, por ejemplo, para la clase smartphone, el modelo entrenado en Dataset-3 muestra una precisión del 57.22 %, recall 82.60 % y F1 67.61 % mientras que para la clase billete el modelo muestra una precisión de 78.83 %, recall 87.80 % y F1 83.07 %. Esto significa que el uso de un gran número de imágenes no implica un mejor aprendizaje sino que imágenes de mayor calidad proporcionan una mejora en el aprendizaje del modelo. El aprendizaje del modelo mejora sustancialmente desde Dataset-3 a Dataset-5 gracias a una mayor calidad y diversidad de las imágenes incluidas en Dataset-5. Los mejores resultados y más equilibrados por clase se obtienen al entrenar el modelo de seis clases, Dataset-5, ya que tiene una mayor calidad y diversidad. En el resto del trabajo utilizaremos Dataset-5 para la evaluación de OVA y OVO ya que es la base de datos que permite al modelo distinguir mejor entre los diferentes objetos.

IV. EVALUACIÓN DE OVA, OVO Y DRCW-OVO

En esta sección se evaluará el rendimiento de los métodos OVA, OVO y DRCW-OVO en nuestro problema de multi-clasificación. Para entrenar estos modelos, se ha usado la base de datos Dataset-5 y para testear las distintas técnicas y evaluarlas, Dataset-Test. Además, compararemos con el modelo entrenado sobre Dataset-5 al tener mejor resultados y ser el modelo de referencia. Primero analizamos OVA en IV-A, OVO en IV-B y DRCW-OVO en IV-C.

IV-A. Estrategia OVA

Para obtener los resultados del método de agregación MAX que vemos en la Tabla III, se han entrenado seis clasificadores y se han testeado sobre el conjunto de test, Dataset-Test. En este se pueden ver las predicciones, y las medidas precisión, recall y F1 obtenidas en cada clase.

Si comparamos los resultados de OVA con el modelo de referencia multiclase, observamos que los resultados son muy similares. El margen de mejora de OVA con respecto al modelo



Tabla III

PARA OVA MAX, PREDICCIONES, PRECISIÓN, RECALL Y F1 DE CADA CLASE, TESTEADO EN DATASET-TEST.

OVA MAX	Background	Billete	Monedero	Pistola	Smartphone	Tarjeta	Precisión	Recall	F1	
Predicción	Background	235	4	8	1	0	95.98 %	83.75 %	89.45 %	
	Billete	5	116	0	1	0	92.80 %	94.30 %	93.54 %	
	Monedero	6	2	90	2	15	0	78.26 %	86.53 %	82.19 %
	Pistola	36	0	1	290	1	2	87.87 %	98.63 %	92.94 %
	Smartphone	13	1	3	0	97	1	84.34 %	84.34 %	84.34 %
	Tarjeta	5	0	2	0	1	58	87.87 %	90.62 %	89.23 %
MEDIA							87.85 %	89.70 %	88.62 %	

de referencia es insignificante. El modelo base entrenado en Dataset-5 obtuvo un precisión de 87,89 %, recall 89,45 % y un F1 88,46 %, mientras que OVA con el método MAX obtuvo un precisión de 87,85 %, recall 89,70 % y F1 88,62 %.

IV-B. Estrategia OVO

Nuestro problema es un problema de seis clases, por lo que OVO tendrá quince clasificadores. En la Tabla IV y para cada subtabla, se muestran los resultados de una estrategia de agregación diferente expresado en términos de predicciones, precisión, recall y F1 obtenidos para el test Dataset-Test.

Tabla IV

PARA CADA MÉTODO DE AGREGACIÓN DE OVO, PREDICCIONES, PRECISIÓN, RECALL Y F1 DE CADA CLASE, TESTEADO EN DATASET-TEST.

OVO VOTE random	Background	Billete	Monedero	Pistola	Smartphone	Tarjeta	Precisión	Recall	F1	
Predicción	Background	337	3	7	1	1	96.29 %	84.02 %	89.87 %	
	Billete	5	115	1	0	2	93.50 %	93.50 %	93.50 %	
	Monedero	7	4	92	3	12	0	77.97 %	82.88 %	
	Pistola	40	0	2	288	3	1	86.23 %	97.96 %	91.72 %
	Smartphone	8	0	2	2	98	2	87.50 %	85.22 %	86.34 %
	Tarjeta	3	1	0	0	1	58	92.06 %	90.62 %	91.34 %
MEDIA							88.92 %	90.00 %	89.27 %	

OVO VOTE by weight	Background	Billete	Monedero	Pistola	Smartphone	Tarjeta	Precisión	Recall	F1	
Predicción	Background	341	4	8	1	1	97.79 %	85.25 %	90.21 %	
	Billete	4	114	1	0	2	94.21 %	92.68 %	93.44 %	
	Monedero	6	4	91	3	11	0	79.13 %	87.50 %	83.11 %
	Pistola	37	0	2	288	3	2	86.75 %	92.68 %	92.01 %
	Smartphone	10	0	2	2	99	1	86.84 %	86.09 %	86.46 %
	Tarjeta	2	1	0	0	1	58	93.55 %	90.62 %	92.06 %
MEDIA							89.38 %	90.02 %	89.55 %	

OVO WV	Background	Billete	Monedero	Pistola	Smartphone	Tarjeta	Precisión	Recall	F1	
Predicción	Background	341	5	8	1	1	95.52 %	85.25 %	90.09 %	
	Billete	4	114	0	0	2	95.00 %	92.68 %	93.83 %	
	Monedero	6	3	92	3	11	0	80.00 %	88.46 %	84.02 %
	Pistola	37	0	2	288	3	2	86.75 %	97.96 %	92.01 %
	Smartphone	10	0	2	2	99	1	86.84 %	86.09 %	86.46 %
	Tarjeta	2	1	0	0	1	58	93.55 %	90.62 %	92.06 %
MEDIA							89.61 %	90.18 %	89.75 %	

OVO LVPC	Background	Billete	Monedero	Pistola	Smartphone	Tarjeta	Precisión	Recall	F1	
Predicción	Background	342	5	9	2	1	2	94.74 %	85.50 %	89.88 %
	Billete	4	114	0	0	3	2	95.80 %	92.68 %	93.83 %
	Monedero	5	3	91	3	10	0	81.25 %	87.50 %	84.26 %
	Pistola	39	0	2	288	5	2	85.71 %	97.96 %	91.43 %
	Smartphone	8	0	2	1	98	1	89.09 %	85.22 %	87.11 %
	Tarjeta	2	1	0	0	1	57	93.44 %	89.06 %	91.20 %
MEDIA							89.87 %	89.65 %	89.62 %	

OVO ND	Background	Billete	Monedero	Pistola	Smartphone	Tarjeta	Precisión	Recall	F1	
Predicción	Background	339	4	7	1	1	1	96.03 %	84.75 %	90.04 %
	Billete	4	114	1	0	0	2	94.21 %	92.68 %	93.44 %
	Monedero	6	4	90	3	11	0	79.35 %	86.54 %	82.97 %
	Pistola	40	0	2	289	3	2	86.01 %	98.30 %	91.75 %
	Smartphone	8	0	4	1	99	1	87.61 %	86.09 %	86.84 %
	Tarjeta	3	1	0	0	1	58	92.06 %	90.62 %	91.34 %
MEDIA							89.15 %	89.83 %	89.33 %	

OVO PC	Background	Billete	Monedero	Pistola	Smartphone	Tarjeta	Precisión	Recall	F1	
Predicción	Background	341	5	8	1	1	1	95.52 %	85.25 %	90.09 %
	Billete	4	114	0	0	0	1	95.80 %	92.68 %	94.21 %
	Monedero	6	3	93	3	12	0	79.49 %	89.42 %	84.16 %
	Pistola	38	0	2	288	3	2	86.49 %	97.96 %	91.87 %
	Smartphone	9	0	1	2	98	1	88.29 %	85.22 %	86.75 %
	Tarjeta	2	1	0	0	1	59	93.65 %	92.19 %	92.91 %
MEDIA							89.87 %	90.45 %	90.00 %	

OVO PE	Background	Billete	Monedero	Pistola	Smartphone	Tarjeta	Precisión	Recall	F1	
Predicción	Background	339	4	7	1	1	1	95.76 %	84.75 %	89.92 %
	Billete	4	114	0	0	2	0	95.00 %	92.68 %	93.83 %
	Monedero	6	4	91	3	11	0	79.13 %	87.50 %	83.11 %
	Pistola	40	0	2	288	3	2	85.97 %	97.96 %	91.57 %
	Smartphone	8	0	4	1	99	1	87.61 %	86.09 %	86.84 %
	Tarjeta	3	1	0	0	1	58	92.06 %	90.62 %	91.34 %
MEDIA							89.26 %	89.93 %	89.43 %	

Como podemos observar en la Tabla IV, OVO PC supera al resto de métodos, alcanzando unos valores medios de precisión del 89,87 %, recall 90,45 % y F1 90,00 %. Además, OVO PC obtuvo el mejor rendimiento sobre todas las estrategias evaluadas, OVA y el modelo multclasificador base. En particular, OVO PC mejoró el modelo base en un 1,98 % en precisión, un 1 % en recall y un 1,54 % en F1.

IV-C. Estrategia DRCW-OVO

DRCW-OVO se evalúa en cuatro valores diferentes de k, el número de vecinos más cercanos que hacen el promedio medio en el valor de la distancia. Los resultados se muestran en la Tabla V. Cada subtabla muestra los resultados de un valor de k diferente expresado en términos de predicciones, precisión, recall y F1 obtenido sobre Dataset-Test.

Tabla V

PARA DISTINTOS VALORES DE K EN DRCW-OVO, PREDICCIONES, PRECISIÓN, RECALL Y F1 DE CADA CLASE, TESTEADO EN DATASET-TEST.

DRCW-OVO k=1	Background	Billete	Monedero	Pistola	Smartphone	Tarjeta	Precisión	Recall	F1	
Predicción	Background	347	5	8	2	1	1	95.33 %	86.75 %	90.84 %
	Billete	2	114	0	0	0	1	97.44 %	92.68 %	95.00 %
	Monedero	5	3	91	3	9	0	81.98 %	87.50 %	84.65 %
	Pistola	35	0	2	288	4	1	87.22 %	97.96 %	92.31 %
	Smartphone	8	0	3	1	100	1	88.50 %	86.96 %	87.23 %
	Tarjeta	3	1	0	0	1	60	92.31 %	93.75 %	93.02 %
MEDIA							90.47 %	90.93 %	90.59 %	

DRCW-OVO k=3	Background	Billete	Monedero	Pistola	Smartphone	Tarjeta	Precisión	Recall	F1	
Predicción	Background	343	5	9	2	1	9	95.01 %	85.75 %	90.14 %
	Billete	2	114	0	0	0	1	97.44 %	92.68 %	95.00 %
	Monedero	5	3	91	3	11	0	80.53 %	87.50 %	83.87 %
	Pistola	39	0	2	288	4	1	86.23 %	97.96 %	91.72 %
	Smartphone	8	0	2	1	98	1	89.09 %	85.22 %	87.11 %
	Tarjeta	3	1	0	0	1	60	92.31 %	93.75 %	93.02 %
MEDIA							90.10 %	90.48 %	90.14 %	

DRCW-OVO k=5	Background	Billete	Monedero	Pistola	Smartphone	Tarjeta	Precisión	Recall	F1	
Predicción	Background	343	5	9	2	1	1	95.01 %	85.75 %	90.14 %
	Billete	2	114	0	0	0	1	97.44 %	92.68 %	95.00 %
	Monedero	5	3	91	3	10	0	81.25 %	87.50 %	84.65 %
	Pistola	39	0	2	288	5	1	85.97 %	97.96 %	91.57 %
	Smartphone	8	0	2	1	98	1	89.09 %	85.22 %	87.11 %
	Tarjeta	3	1	0	0	1	60	92.31 %	93.75 %	93.02 %
MEDIA							90.18 %	90.48 %	90.19 %	

DRCW-OVO k=7	Background	Billete	Monedero	Pistola	Smartphone	Tarjeta	Precisión	Recall	F1	
Predicción	Background	341	5	9	2	1	1	94.99 %	85.25 %	89.86 %
	Billete	3	114	0	0	0	1	96.61 %	92.68 %	94.61 %
	Monedero	5	3	91	3	9	0	81.98 %	87.50 %	84.65 %
	Pistola	40	0	2	288	5	1	85.71 %	97.96 %	91.43 %
	Smartphone	8	0	2	1	99	1	89.19 %	86.09 %	87.61 %
	Tarjeta	3	1	0	0	1	60	92.31 %	93.75 %	93.02 %
MEDIA							90.13 %	90.54 %	90.20 %	

Como podemos observar en esta Tabla V, el método DRCW-OVO con valor k de uno, obtuvo el mejor resultado sobre todos los OVO evaluados, OVA y el modelo multclasificador base. Esta configuración de DRCW-OVO ha obtenido un precisión media de 90.47 %, recall 90.93 % y F1 90.59 %. Esta técnica mejoró el modelo multclasificador base en un 2,58 % en precisión, 1,48 % recall y 2,13 % F1. Esta mejora, con el alto valor de las medidas obtenidas, puede suponer un incremento importante en los resultados de la clasificación.

V. CONCLUSIONES Y TRABAJO FUTURO

En este trabajo, proponemos el uso de técnicas de binarización como OVA y OVO para mejorar los resultados de una CNN normal en la tarea de clasificación. Por esta razón, y con la experiencia en este tipo de situaciones, utilizamos un conjunto de datos formado por armas de fuego y otros objetos que se manejan de forma similar. Los pasos realizados fueron, (1) incluir las clases de objetos que pueden suponer una confusión para el modelo como smartphone, billete, monedero y tarjeta, y añadir imágenes con más calidad y contexto a la base de datos utilizando distintas cámaras y contextos, (2) utilizar un nuevo modelo de clasificación como ResNet-101 y (3) mejorar la robustez utilizando técnicas de *machine learning* como OVA, OVO y DRCW-OVO.

Los resultados obtenidos del modelo entrenado en Dataset-5 y testeado sobre Dataset-Test obtienen una mejora de rendimiento de 35.7 % en precisión, 8.7 % recall y 28.05 % F1. Dataset-5 se usa para entrenar los modelos OVA, OVO y DRCW-OVO y los resultados del mejor modelo fueron de

90.47 % en precisión, recall de 90.93 % y F1 de 90.59 % para la técnica DRCW-OVO $k=1$. En resumen, hemos alcanzado una mejora del 2,58 % en precisión, del 1,48 % en recall y del 2,13 % en F1. Esto significa que, DRCW-OVO obtiene el valor más alto sobre todas las estrategias evaluadas.

Nuestro trabajo futuro consistirá en estudiar otros objetos que impliquen conflictos. Además, se hará un filtro para las instancias ruidosas que pueden causar confusión en los modelos CNN.

AGRADECIMIENTOS

Este trabajo esta parcialmente respaldado por el Ministerio de Ciencia y Tecnología de España, en el proyecto TIN2017-89517-P y por la Junta de Andalucía en el proyecto P11-TIC-7765. Siham Tabik fue apoyada por el programa Ramon y Cajal (RYC-2015-18136) y Francisco Pérez por el Programa de personal técnico financiado por el programa operativo de empleo juvenil. Las Titan X Pascal usadas para esta investigación han sido donadas por NVIDIA Corporation.

REFERENCIAS

- [1] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 248–255.
- [2] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.
- [3] G. Flitton, T. P. Breckon, and N. Megherbi, "A comparison of 3d interest point descriptors with application to airport baggage object detection in complex ct imagery," *Pattern Recognition*, vol. 46, no. 9, pp. 2420–2436, 2013.
- [4] A. Glowacz, M. Kmiec, and A. Dziech, "Visual detection of knives in security applications using active appearance models," *Multimedia Tools and Applications*, vol. 74, no. 12, pp. 4253–4267, 2015.
- [5] R. K. Tiwari and G. K. Verma, "A computer vision based framework for visual gun detection using harris interest point detector," *Procedia Computer Science*, vol. 54, pp. 703–712, 2015.
- [6] I. Uroukov and R. Speller, "A preliminary approach to intelligent x-ray imaging for baggage inspection at airports," *Signal Processing Research*, vol. 4, pp. 1–11, 2015.
- [7] Z. Xiao, X. Lu, J. Yan, L. Wu, and L. Ren, "Automatic detection of concealed pistols using passive millimeter wave imaging," in *Imaging Systems and Techniques (IST), 2015 IEEE International Conference on*. IEEE, 2015, pp. 1–4.
- [8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.
- [9] R. Olmos, S. Tabik, and F. Herrera, "Automatic handgun detection alarm in videos using deep learning," *Neurocomputing*, 2017.
- [10] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, and F. Herrera, "An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes," *Pattern Recognition*, vol. 44, no. 8, pp. 1761–1776, 2011.
- [11] M. Galar, A. Fernández, E. Barrenechea, and F. Herrera, "Drcw-ovo: distance-based relative competence weighting combination for one-vs-one strategy in multi-class problems," *Pattern recognition*, vol. 48, no. 1, pp. 28–42, 2015.
- [12] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.
- [13] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.
- [14] J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks," in *Advances in neural information processing systems*, 2016, pp. 379–387.
- [15] A. Rocha and S. K. Goldenstein, "Multiclass from binary: Expanding one-versus-all, one-versus-one and ecoc-based approaches," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 2, pp. 289–302, 2014.
- [16] M. Yu, L. Gong, and S. Kollias, "Computer vision based fall detection by a convolutional neural network," in *Proceedings of the 19th ACM International Conference on Multimodal Interaction*. ACM, 2017, pp. 416–420.
- [17] X. Chen, T. Fang, H. Huo, and D. Li, "Measuring the effectiveness of various features for thematic information extraction from very high resolution remote sensing imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 9, pp. 4837–4851, 2015.
- [18] K. Öztürk and M. B. Yilmaz, "A comparison of classification approaches for deep face recognition," in *Computer Science and Engineering (UBMK), 2017 International Conference on*. IEEE, 2017, pp. 227–232.
- [19] H. Lei and V. Govindaraju, "Half-against-half multi-class support vector machines," in *International Workshop on Multiple Classifier Systems*. Springer, 2005, pp. 156–164.
- [20] P. Clark and R. Boswell, "Rule induction with cn2: Some recent improvements," in *European Working Session on Learning*. Springer, 1991, pp. 151–163.
- [21] S. Knerl, L. Personnaz, and G. Dreyfus, "Single-layer learning revisited: a stepwise procedure for building and training a neural network," in *Neurocomputing*. Springer, 1990, pp. 41–50.
- [22] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, and F. Herrera, "Aggregation schemes for binarization techniques methods' description," *Pamplona, Spain*, 2011.
- [23] J. Friedman, "Another approach to polychotomous classification," Technical report, Department of Statistics, Stanford University, Tech. Rep., 1996.
- [24] E. Hüllermeier and K. Brinker, "Learning valued preference structures for solving classification problems," *Fuzzy Sets and Systems*, vol. 159, no. 18, pp. 2337–2352, 2008.
- [25] J. C. Hühn and E. Hüllermeier, "Fr3: A fuzzy rule learner for inducing reliable classifiers," *IEEE Transactions on Fuzzy Systems*, vol. 17, no. 1, pp. 138–149, 2009.
- [26] S. Orlovsky, "Decision-making with a fuzzy preference relation," in *Readings in Fuzzy Sets for Intelligent Systems*. Elsevier, 1993, pp. 717–723.
- [27] A. Fernández, M. Calderón, E. Barrenechea, H. Bustince, and F. Herrera, "Enhancing fuzzy rule based systems in multi-classification using pairwise coupling with preference relations," *EUROFUSE*, vol. 9, pp. 39–46, 2009.
- [28] T. Hastie and R. Tibshirani, "Classification by pairwise coupling," in *Advances in neural information processing systems*, 1998, pp. 507–513.
- [29] T.-F. Wu, C.-J. Lin, and R. C. Weng, "Probability estimates for multi-class classification by pairwise coupling," *Journal of Machine Learning Research*, vol. 5, no. Aug, pp. 975–1005, 2004.
- [30] O. Pele and M. Werman, "The quadratic-chi histogram distance family," in *European conference on computer vision*. Springer, 2010, pp. 749–762.
- [31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.
- [32] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: A system for large-scale machine learning," in *OSDI*, vol. 16, 2016, pp. 265–283.