PAR FAS 2007-2013 - LINEA A

SmartNEWS

Social sensing for breaking news

Documento D3.3

Visual Content Mining

Nome del progetto	Social sensing for breaking news
Acronimo	SMART NEWS
Data avvio progetto	14/03/2016
Durata del progetto	24 mesi (prorogato a 30)
Data	13 Marzo 2018 [24]
Stato del documento	Prima Versione
Responsabile	ISTI-CNR
Editore	Lucia Vadicamo (ISTI-CNR)
Autori	Giuseppe Amato (ISTI-CNR), Fabio Carrara (ISTI-CNR), Fabrizio Falchi (ISTI-CNR), Claudio Gennaro (ISTI-CNR), Lucia Vadicamo (ISTI-CNR)
Contributi	Hyperborea Srl, IIT-CNR, ISTI-CNR, ILC-CNR
Revisori	Tiziano Fagni (IIT-CNR)

Descrizione

Il presente deliverable D3.3 "Visual Content Mining" ha lo scopo di descrivere e documentare le attività di visual content mining portate avanti come parte dell'obiettivo operativo 3 "Social Media Analysis/Mining".

In particolare, questo documento descrive lo stato dell'arte e le tecniche adottate o sviluppate in SmartNews per l'analisi automatica delle immagini al fine di estrarre informazioni che ne permettano la loro descrizione automatica, classificazione e ricerca. Tali analisi verranno integrate nel News Management tool per l'analisi delle immagini raccolte dal sistema (attività 3.1 "Data Collection") fornendo agli utenti della piattaforma degli strumenti innovativi per l'analisi dei dati e l'arricchimento delle informazioni raccolte su una notizia monitorata.

Sommario

Glossario dei termini

ABBREVIAZIONE	DEFINIZIONE
API	Application Programming Interface
BMM-FV	Bernoulli Mixture Model- Fisher Vector. Tecnica di aggregazione di feature binarie.
BoW	Bag-of-Word. Tecnica di aggregazione di feature locali di immagini.
CBIR	Content Based Image Retrieval. Sistemi capaci di archiviare e reperire le immagini utilizzandone il loro "contenuto visivo", ossia senza l'uso di tag od altri metadati.
CNN	Convolutional Neural Network. Una tipologia di reti neurali utilizzate spesso per l'estrazione di informazioni della immagini.
FV	Fisher Vector. Tecnica di aggregazione di feature locali di immagini.
PBI	Permutation-Based Indexing. Classe di tecniche di indicizzazione per ricerche approssimate.
R-MAC	Regional Maximum Activations of Convolutions. Una particolare classe di deep feature.
STR	Surrogate Text Representation. Tecnica di codifica testuale di una permutazione per l'utilizzo di indici testuali.
T4SA	Twitter for Sentiment Analysis. Dataset creato nel progetto SmartNews per l'allenamento di un classificatore del sentimento di immagini.
VLAD	Vector of Locally Aggregated Descriptors. Tecnica di aggregazione di feature locali di immagini.

1 Introduzione

Tra i requisiti del News Management tool di SmartNews vi è l'analisi automatica dei dati raccolti su eventi o news di interesse al fine di fornire all'utente un quadro generale della situazione monitorata. Questo deliverable si concentra sui metodi e le tecniche utilizzate in SmartNews per l'analisi automatica delle immagini, ovvero descrive l'attività 3.3 "Visual content mining" che è parte integrante dell'Obiettivo Operativo 3 "Social Media Analysis/Mining".

Infatti, questa attività si occupa dell'analisi automatica delle immagini raccolte nell'attività 3.1 "Data Collection" (si veda il Deliverable D3.1) con gli obiettivi di creare collegamenti semantici fra le immagini sulla base del solo contenuto visuale, nonché raggruppare e ricercare le immagini visivamente simili fra loro.

Tali obiettivi sono stati raggiunti mediante attività di ricerca scientifica e sviluppo software per

- l'**annotazione testuale**, ossia la classificazione semantica del contenuto delle immagini che ne descriva gli oggetti, gli attributi, le azioni e le scene in esse rappresentati;
- la **ricerca per immagine**, ossia la ricerca in archivi visuali (anche non etichettati) utilizzando un'immagine come *query*, basata sull'estrazione di descrittori (*feature*) dalle immagini, e sulla loro indicizzazione e ricerca;
- il **clustering**, ossia il raggruppamento delle immagini in insiemi visualmente e semanticamente coerenti per molteplici fini, quali l'analisi multimediale di particolari gruppi o sorgenti social, l'individuazione di immagini uniche, duplicate o quasi-duplicate e l'aiuto alla costruzione di sistemi di visualizzazione e navigazione di immagini;
- l'**analisi del sentimento**, ossia l'individuazione automatica della polarità (positiva, negativa o neutra) del sentimento delle immagini al fine di di fornire all'utente un ulteriore ed innovativo strumento di analisi dei dati visuali raccolti su una news o evento di interesse.

L'attività di "Visual content mining", inizialmente programmata da mese 6 a mese 13, è stata estesa fino al termine del progetto in modo da fare costantemente fronte all'evoluzione scientifica e tecnologica che questo settore di ricerca sta vivendo. Infatti, i recenti sviluppi nel campo dell'intelligenza artificiale ed in particolare del *deep learning* hanno portato significativi miglioramenti rispetto allo stato dell'arte nell'analisi e mining delle immagini. Alcuni approcci ed algoritmi di Computer Vision basati su feature locali di immagini e loro aggregazioni (come *VLAD* e *Fisher Vector)* , il cui uso era previsto nella Scheda Tecnica di Progetto, sono stati investigati ed utilizzati solo nel primo anno di attività del progetto e successivamente sostituiti da efficaci soluzioni basate su deep learning, quali l'uso di *deep feature* (si veda la Sezione 2.1 per ulteriori dettagli . Inoltre, sono state sviluppate funzionalità innovative, come la classificazione del sentimento delle immagini (la Sezione 5), che inizialmente non erano previste nel progetto. La scelta di affrontare la tematica dell'analisi del sentimento delle immagini è stata suggerita da notevoli fattori quali la rilevanza scientifica di questo tipo di analisi, la cospicua quantità di dati multimediali resi disponibili nell'ambito del progetto e la disponibilità di un classificatore del sentimento per il testo realizzato nell'Attività 3.2 "Text Mining".

L'output dell'attività 3.3, oltre alla produzione di questo documento, include la realizzazione dei moduli di Visual Content mining (le cui API sono descritte nel Deliverable D4.3 "Definizione delle APIs" e riassunte nelle sezione 6 di questo documento) e la pubblicazione di numerosi articoli scientifici su riviste e conferenze internazionali [Amato et al. 2016a, Amato et al. 2016b, Amato et al. 2016c, Amato et al. 2016d, Amato et al. 2016e, Amato et al. 2017, Carrara et al. 2017b, Conner et al. 2017, Conner et al. 2018, Vadicamo et al. 2017].

1.1 Struttura del documento

Il presente deliverable è organizzato in sette sezioni, inclusa la prima sezione di introduzione. La **Sezione 2** introduce il problema della ricerca per immagini, e riassume lo stato dell'arte e le tecniche utilizzate in SmartNews per l'estrazione

automatica di informazioni dalle immagini che ne permetta la loro rappresentazione e ricerca. La **Sezione 3** è dedicata alle tecniche di classificazione ed annotazione automatica. La **Sezione 4** presenta le tecnica di clustering usata nel progetto per il raggruppamento di immagini visivamente simili e l'identificazione di immagini duplicate. La tematica del riconoscimento visuale del sentimento di una immagine è affrontata nella **Sezione 5**. La **Sezione 6** fornisce una breve descrizione delle API dei moduli di Content Visual Mining e, infine, la **Sezione 7** conclude il documento.

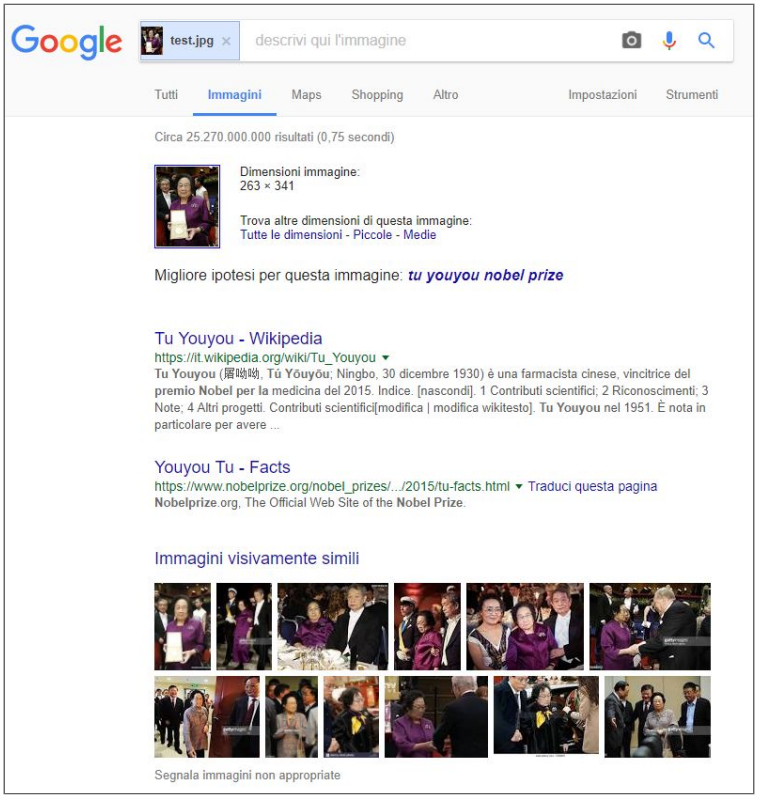
2 Ricerca per immagine

Le immagini hanno un elevato potere comunicativo che risiede soprattutto nella possibilità di trasmettere e comprendere un messaggio od un concetto nel più breve tempo possibile. Si osservi, per esempio, la Figura 1 dove uno stesso concetto è rappresentato sia visivamente che testualmente. Le reazioni della nostra mente di fronte ad uno stimolo puramente visivo o testuale sono molto diverse. Notoriamente la mente umana elabora le immagini molto più velocemente dei testi scritti, e non solo, le immagini catturano facilmente la nostra attenzione e permettono di trascendere barriere linguistiche e culturali. Per questo le immagini sono da sempre considerate uno dei canali di comunicazione più efficaci per condividere idee ed informazioni. Oggigiorno milioni di immagini vengono condivise quotidianamente sui social media. Basti pensare ad Instagram che ha attualmente ha più di 700 milioni di utenti attivi ed un totale di 34.7 miliardi di immagini condivise, registrando una media di 15 milioni di upload ogni giorno.

Figura 1: La mente umana processa le immagini molto più velocemente dei testi

La dilagante diffusione delle fotocamere digitali, come quelle integrate in cellulari e tablet, e la disponibilità di numerosi sistemi di archiviazione tramite Internet hanno favorito e favoriscono tutt'oggi una massiccia produzione di contenuti multimediali, ponendo al contempo il problema della gestione di grandi archivi visuali. Inoltre, il trend seguito dagli utenti dei social media, abituati a scattare e condividere le proprie foto senza descriverne propriamente il contenuto, rispecchia la diffusa mancanza di metadati associati alle immagini. Tali fattori hanno reso opportuno lo studio e lo sviluppo di sistemi di *Content Based Image Retrieval* (CBIR) capaci di archiviare e reperire le immagini utilizzandone il loro "contenuto visivo", ossia senza l'uso di tag od altri metadati ad essi associati. I sistemi di CBIR permettono quindi di effettuare ricerche in archivi visuali (anche non etichettati) utilizzando un'immagine come *query*, che probabilmente è la via più immediata e semplice per ottenere informazioni su un oggetto visibile o di interesse. Immaginate, ad esempio, di aver visto su un social network il seguente post:

e di voler scoprire chi è la persona raffigurata nella foto e in quale evento è stata scatatta la foto. Ottenere le informazioni volute mediante un classico motore di ricerca testuale (come ad esempio www.google.it) non è immediato e richiede sia una forte astrazione logica che la capacità di formulare una query altamente rilevante con il contenuto della foto osservata. Ad esempio, si dovrebbe riconoscere o dedurre che la medaglia d'oro nella foto è quella di un premio Nobel, e probabilmente sfruttare il fatto che la persona premiata sia una donna asiatica. Una query del tipo "Premio Nobel donna asiatica" permette di trovare tra i primi, ma non primissimi, risultati di Google le informazioni volute, richiedendo quindi un ulteriore effort dell'utente nell'analisi dei risultati. Non sarà quindi del tutto immediato scoprire che si tratta di Tu Youyou, che nel 2015 ha vinto il premio Nobel per la medicina grazie alle sue ricerche sulla malaria.
I motori di ricerca per immagini (come ad esempio https://images.google.com/), invece, mediante il semplice upload della foto d'interesse permettono di ottenere le informazioni volute in maniera più immediata, come mostrato nel seguente screenshot di Google Images:



L'immediatezza e la semplicità di utilizzo che caratterizza la ricerca per immagini è alla base del suo successo ed uso in moltissime applicazioni. Per esempio la Museum Mobile Guide (MMG) della Pinacoteca di Brera[1] è una app che permette ai visitatori della Pinacoteca di ottenere informazioni su un dipinto, o qualsiasi opera d'arte della galleria, semplicemente scattando una foto dell'oggetto di interesse con il proprio smartphone.

A livello scientifico, la ricerca basata sul contenuto delle immagini richiede di affrontare numerosi problematiche quali la scelta di un'opportuna rappresentazione matematica del contenuto delle immagini e di come memorizzare e organizzare i dati cosicché la ricerca possa essere effettuata in maniera efficiente e scalabile. L'idea alla base di tali tecnologie è quella di rappresentare le immagini mediante dei descrittori numerici (feature) che possano essere comparati tra loro e tali che immagini con contenuto visivo simile siano associate a descrittori simili dal punto di vista matematico. Tipicamente le feature associate alle immagini vivono in uno spazio metrico, ossia in uno spazio

[1] Museum Mobile Guide - Pinacoteca di Brera. https://mmg.inera.it/frontend/pinacoteca-di-brera-2.

matematico dove è possibile calcolare la distanza tra due generici punti. Minore è la distanza tra due feature, maggiore è la loro similarità.

Riuscire a fornire automaticamente una rappresentazione delle immagini che rispecchi la similarità visuale è un problema che presenta varie criticità, prima tra tutte il superamento del *semantic gap* [Smeulders et al. 2000], ossia del divario tra il limitato potere descrittivo delle feature di immagini e la ricchezza dei concetti rappresentati nelle immagini stesse. Decenni di ricerca su tematiche quali relevance feedback, annotazione automatica e le recenti tecnologie basate sul deep learning hanno aiutato a mitigare questo problema [Zhou & Huang 2003, Datta et al. 2005,Hare et al. 2006, Liu et al. 2007, Guo et al. 2016, Li et al. 2016].

In Figura 2 sono rappresentate tre fasi fondamentali di un generico sistema CBIR: 1) la scelta di un'opportuna rappresentazione matematica delle immagini che ne permetta il confronto per similarità (*feature extraction*); 2) l'indicizzazione di tali rappresentazioni (*indexing*); 3) la fase di ricerca data una immagine come query (*searching*).

Nelle sezioni successive verranno presentati lo stato dell'arte e gli approcci usati nel progetto SmartNews per la rappresentazione delle immagini, l'indicizzazione e la ricerca visuale.

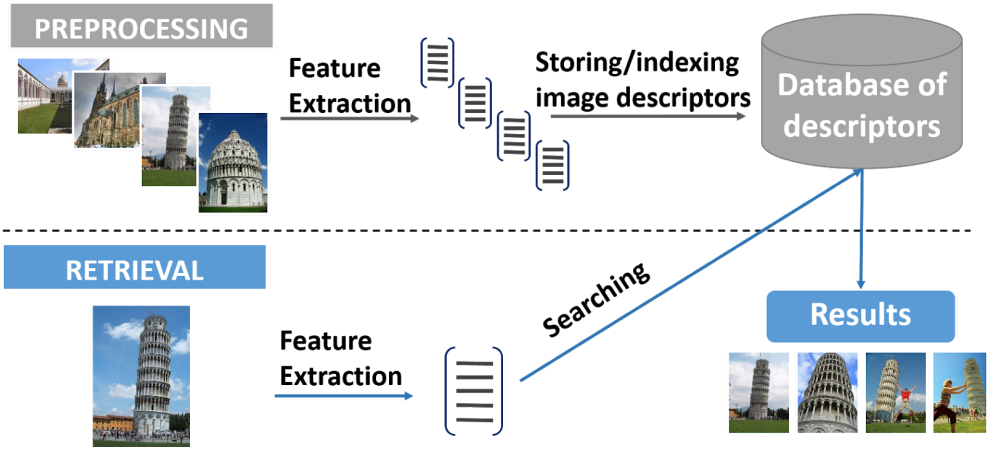


Figura 2: Fasi fondamentali di un generico sistema di CBIR

2.1 Rappresentazione delle immagini

Alla base di un qualsiasi sistema CBIR c'è la scelta di quali feature utilizzare per la rappresentazione del contenuto visuale delle immagini. Questo significa che ciascuna immagine è rappresentata da una o più feature che catturano determinate caratteristiche delle immagini. Le feature devono essere descrittive e discriminative. Inoltre, la richiesta generale è che esse siano invarianti, o quanto meno robuste, a variazioni delle immagini come rotazioni, traslazioni, riscaldamento, cambiamento di luminosità o punto di vista. Ciascuna feature è rappresentata da descrittore numerico, che nella maggior parte dei casi è un vettore o un elemento di uno spazio matematico su cui sia possibile effettuare delle comparazioni di similarità. Infatti, l'idea principale su cui si basano i sistemi di CBIR è che tanto più simili sono le feature estratte da due immagini tanto più probabile è che le due immagini abbiano un contenuto visuale simile. Di seguito, così come accade spesso nell'ambito della Computer Vision, si userà il termine "feature" per indicare non solo la tipologia di caratteristiche visuali estratte dalle immagini ma anche il descrittore (ossia il vettore) usato per la rappresentazione numerica della feature.

Dato che la scelta delle feature da usare per descrivere le immagini è cruciale per la ricerca per similarità visuale, molte tipologie di feature sono state proposte ed analizzate nella letteratura scientifica dell'ultimo decennio, specialmente nell'ambito della Computer Vision. I primi approcci di CBIR utilizzavano soprattutto feature globali, quali descrittori del colore, texture e forma [Deng & Manjunath 2001,Manjunath et al. 2002, Park et al. 2002, Kodituwakku & Selvarajah 2004, Mehrotra & Gary 1995]. Il vantaggio principale delle feature globali è che sono veloci da estrarre e

comparare. Tuttavia, esse hanno dimostrato di avere un basso potere discriminativo e ad oggi sono considerate troppo "rigide" per la rappresentazione delle immagini.

L'intuizione che diverse regioni di un'immagine possano portare un diverso contributo alla descrizione del contenuto globale dell'immagine ha portato alla definizione e diffusione delle cosiddette feature locali (*local feature)*, come SIFT [Lowe 2004] e SURF [Bay et al 2006]. Ciascuna feature locale fornisce una descrizione di una piccola regione dell'immagine, delimitata in un intorno di un punto di interesse (*keypoint*). I punti di interesse sono automaticamente selezionati in zone dell'immagine con un'elevata variazione (come i *corner* ed i *blob*) così che essi possano essere identificati in maniera ripetibile. Questa tipologia di feature permettono di confrontare efficacemente le strutture locali di due immagini; tuttavia, tale confronto non è efficiente poiché ha un elevato costo computazionale sia in termini di calcolo che di occupazione di memoria. Infatti, ciascuna immagine è rappresentata da migliaia di descrittori locali (vettori) e per decidere se due immagini sono simili, nel peggiore dei casi, bisogna confrontare ciascun descrittore della prima immagine con ciascun descrittore della seconda immagine. Anche se si utilizzassero strutture dati, come il kd-tree [Friedman et al. 1977], molto efficienti per il matching delle feature, il costo computazionale sarebbe comunque limitante qualora si vogliano effettuare ricerche per similarità visuale su grandi database o qualora tale ricerche debbano essere supportate su dispositivi con ridotte risorse computazionali come smartphone e tablet.

Per superare tali limiti, in letteratura, sono state seguite principalmente due direzioni. Da un lato sono stati introdotti i metodi di *aggregazione di feature locali*, che permettono di rappresentare ciascuna immagine mediante un unico descrittore numerico. Dall'altro sono state introdotte le *feature locali binarie* che rispetto alle feature non-binarie hanno una minore occupazione di memoria e sono più veloci da calcolare e confrontare.

Le tecniche di aggregazione, come Bag-of-Word (BoW) [Sivic & Zisserman 2003], VLAD [Jégou et al 2010a] e Fisher Vector (FV) [Perronnin & Dance 2007], permettono di riassumere le informazioni contenute nei descrittori locali di un'immagine mediante il calcolo di opportune statistiche sulla loro distribuzione. I metodi di aggregazione hanno dimostrato un duplice vantaggio: da un lato, sono risultati particolarmente efficaci in task come la classificazione e l'image retrieval, preservando il potere discriminativo delle feature locali [Jégou et al. 2010b, Perronnin et al. 2010a, Perronnin et al. 2010a]; dall'altro, hanno permesso di ridurre il costo della comparazione tra immagini in quanto ciascuna immagine è rappresentata da un unico descrittore anziché da migliaia di descrittori. Ne segue che l'uso delle tecniche di aggregazione in combinazione all'utilizzo di opportuni indici di ricerca approssimata consentono di scalare la ricerca per immagini su larga scala.

Le feature locali binarie, come le ORB [Rublee et al. 2011], sono state proposte per far fronte alla necessità di avere feature locali che siano compatte e veloci da calcolare, caratteristiche necessarie per applicazioni che richiedono l'analisi di grandi archivi visuali o tempi di risposta fluidi e naturali anche su dispositivi mobili. Per fare un esempio, il processo di estrazione di circa 2000 ORB da una immagine richiede circa 26 millisecondi, mentre l'estrazione di circa 2000 feature locali di tipo SIFT richiederebbe circa 1.2 secondi [Heinly et al 2012]. I tempi di estrazione delle feature locali binarie, infatti, sono fino a due ordini di grandezza più piccoli rispetto alle feature locali non-binarie. Esse sono anche più veloci da confrontare mediante l'uso distanza di Hamming che può essere impiegata in quanto il descrittore associato a ciascuna feature è un vettore binario.

Tuttavia, come nel caso delle feature non-binarie, la rappresentazione delle immagini mediante feature locali binarie risulta ancora limitante per la fase di confronto tra immagini e ricerca in database di grandi dimensioni, poiché ciascuna immagine è rappresentata da migliaia di feature e quindi da migliaia di vettori numerici. Inoltre, le tecniche di aggregazione BoW, VLAD e FV sono state definite assumendo che i descrittori da aggregare siano a valori reali, per cui le versioni originali di tali tecniche non sono del tutto adatte all'aggregazione dei descrittori di tipo binario. Per far fronte a questa limitazione, recentemente sono stati proposte alcune estensioni di tali metodi per l'aggregazione di descrittori binari [Galvez-Lopez & Tardos 2011, Grana et al. 2013, Zhang et al. 2013, Van Opdenbosch et al. 2014, Lee et al. 2015, Uchida et al. 2016].

All'interno del progetto SmartNews, al fine di individuare l'approccio con il miglior compromesso tra efficacia ed efficienza, è stata svolta un'ampia analisi comparativa delle tecniche di aggregazione di feature binare proposte in letteratura. Inoltre, è stata proposta una nuova estensione delle tecnica FV, denominata BMM-FV, che per l'approssimazione della distribuzione dei descrittori binari utilizza una *Bernoulli Mixture Model* anziché una *Gaussian Mixture Model* usata, invece, nella versione originale del FV [Perronnin & Dance 2007]. Il metodo proposto in Smartnews ha mostrato i migliori risultati per il retrieval su due dataset di benchmark (Inria Holidays [Jégou et al. 2008]

ed Oxford 5k [Philbin et al. 2007]), come mostrato in Figura 3. I risultati di tale attività di ricerca sono descritti nella seguente pubblicazione su rivista:

> Amato, G., Falchi, F., & Vadicamo, L. (2016). Aggregating binary local descriptors for image retrieval. *Multimedia Tools and Applications*, 1-31.

Table 3 Performance evaluation of various aggregation methods applied on ORB binary features

Method	Local Feature	Learning method	K	dim	mAP	
					Holidays	Oxford5k
BoW	ORB	*k*-means	20,000	20,000	44.9	22.2
BoW	ORB	*k*-majority	20,000	20,000	44.2	22.8
BoW	ORB	*k*-medoids	20,000	20,000	37.9	18.8
VLAD	ORB	*k*-means	64	16,384	47.8	23.6
VLAD	ORB	*k*-means	64	PCA→ 1,024	46.0	23.2
VLAD	ORB	*k*-means	64	PCA→ 128	30.9	19.3
VLAD	ORB	*k*-majority	64	16,384	32.4	16.6
VLAD	ORB	*k*-medoids	64	16,384	30.6	15.6
FV	ORB	GMM	64	16,384	42.0	20.4
FV	ORB	GMM	64	PCA→ 1,024	42.6	20.3
FV	ORB	GMM	64	PCA→ 128	35.5	19.6
FV	ORB	BMM	64	16,384	49.6	**24.3**
FV	ORB	BMM	64	PCA→ 1,024	**51.3**	23.4
FV	ORB	BMM	64	PCA→ 128	44.6	19.1
No-aggr.	ORB	–	–	–	38.1	31.7

Figura 3:Risultati del confronto delle tecniche di aggregazione di feature binarie su due dataset di benchmark per l'image retrieval. La tecnica BMM-FV è quella che ha ottenuto le migliori performace. Per ulteriori dettagli si rimanda a [Amato et al. 2016a].

Recentemente, gli sviluppi nel campo dell'intelligenza artificiale e in particolare del *deep learning* [LeCun et al. 2015] hanno portato significativi miglioramenti rispetto allo stato dell'arte in numerosi task visivi, tra cui la ricerca per similarità [Razavian et al. 2014]. La popolarità degli approcci basati sulle feature locali di immagini è stata quindi man mano offuscata dai recenti approcci basati sul deep learning. In particolare le feature di immagini ottenute mediante l'uso di deep *Convolutional Neural Networks* (CNN) sono risultate molto più performanti delle aggregazioni di local feature, come FV e VLAD, per la classificazione, il retrieval e il riconoscimento visuale. Per questo motivo le tecniche basate sulle feature locali che erano state adottate all'inizio del progetto sono state accantonate a favore dei recenti approcci basati sul deep learning.

Con il termine "deep learning" si indica un campo dell'intelligenza artificiale che studia algoritmi di apprendimento automatico basati su vari livelli di astrazione nella rappresentazione dei dati. Come riportato da LeCun et al. (2015) sulla rivista Nature:

"*Deep-learning methods are representation-learning methods with multiple levels of representation, obtained by composing simple but non-linear modules that each transform the representation at one level (starting with the raw input) into a representation at a higher, slightly more abstract level. With the composition of enough such transformations, very complex functions can be learned.* "

Per cui le tecniche di deep learning permettono di apprendere una gerarchia di feature dai dati, dove l'aspetto fondamentale è che queste feature non sono non state progettate da ingegneri umani (come le local feature e le loro aggregazioni) ma vengono apprese direttamente dai dati utilizzando approcci di machine learning.

Le deep CNN sono un particolare caso di reti neurali, che hanno avuto un grande successo in applicazioni per l'analisi automatica del contenuto delle immagini. Esse sono composte da numerosi *layer*: un layer di *input*, alcuni layer intermedi (*hidden layer*) e un layer di *output*. Ciascun layer intermedio estrae alcune informazioni dal dato di input mediante delle operazioni matematiche che producono un output che viene dato a sua volta come input al layer successivo; in altre parole le CNN sono reti di tipo "feed-forward". Le operazioni effettuate in ciascun layer sono di tipo non lineare e sono determinate da un insieme di parametri vengono appresi nella fase di addestramento della rete neurale. Nel caso dell'apprendimento supervisionato, per poter allenare una rete neurale di tipo "deep" serve una grande quantità di dati etichettati. Una volta la rete è stata addestrata può essere utilizzata su nuovi dati che la rete non ha mai visto prima. La tipologia del risultato che viene restituito dal layer di output della rete dipende dal task utilizzato in fase di addestramento, come ad esempio la classificazione o la regressione.

Tuttavia l'aspetto interessante è che oltre all'output finale restituito dalla rete, anche gli output dei layer intermedi forniscono una rappresentazione (con vari livelli di astrazione) dei dati in ingresso. Ad esempio, gli output dei layer intermedi di una CNN pre-allenata per il task di classificazione visuale forniscono delle rappresentazioni vettoriali (feature) dell'immagine di input che risultano particolarmente efficaci per il retrieval ed il riconoscimento visuale [Razavian et al. 2014, Babenko et al. 2014]. Questi vettori di attivazione dei layer intermedi di una rete vengono chiamati *deep feature*, o *CNN feature* nel caso in cui la rete utilizzata sia di tipo convoluzionale. Le feature ottenute nei primi layer contengono caratteristiche generiche e meno astratte di quelle degli ultimi layer che, invece, sono maggiormente legate al task di usato in fase di allenamento della rete.

Diverse *deep feature* si differenziano tra loro per il task per cui la rete da cui provengono è stata allenata. Oltre alle *deep feature* estratte da classificatori di immagini, che risultano essere buone rappresentazioni ad alto livello semantico del contenuto dell'immagine, sono state studiate anche le rappresentazioni interne di modelli allenati fin dal principio per lo scopo di image retrieval, più mirate al mantenimento di informazioni dell'immagine di basso livello. Tra le più efficaci in letteratura figurano le *feature* R-MAC (Regional Maximum Activations of Convolutions), un particolare insieme di *deep feature* sviluppate negli ultimi anni [Tolias et al. 2015, Gordo et al. 2016] che sono il risultato di aggregazioni di feature convoluzionali estratte da più regioni di un'immagine. Uno dei vantaggi di questo tipo di rappresentazioni è che l'intero processo di estrazione e aggregazione è implementato completamente con un'architettura di rete neurale convoluzionale ottimizzata per lo scopo di *image retrieval*. Questo tipo di rappresentazione risulta essere quindi molto efficace in questo campo, specialmente nell'*instance image retrieval*, ovvero la ricerca di immagini che contengono particolari istanze di oggetti.

Nonostante features come R-MAC risultino essere lo stato dell'arte nell'instance image retrieval, nell'ambito di SmartNews siamo interessati a una rappresentazione dell'immagine con un alto livello semantico. Lo scopo principale è creare collegamenti semantici fra immagini sulla base del solo contenuto visuale per mettere in relazione tra di loro foto che condividono il soggetto della foto (un luogo, una persona, un particolare evento, un oggetto, etc...), la tipologia di fotografia (ritratto, panorama, immagini artificiale, gruppi di persone, incidenti, etc...), l'immagine stessa (nel caso un'immagine venga postata più volte modificata). Un alto livello di astrazione nella rappresentazione delle immagini permette uno studio più ampio tra le interazioni dei vari media e dei canali di informazione monitorati. Dunque nelle scelte di progetto si è optato per deep features estratte da annotatori generici di immagini, non solo per le proprietà delle loro rappresentazioni interne, ma anche per motivi di efficienza: gli stessi annotatori usati per estrarre le rappresentazioni interne sono utilizzati anche per annotare le immagini con tag generici; questo permette di ottenere entrambe le informazioni con una sola operazione di forward della rete neurale, diminuendo il carico computazionale.

La rete utilizzata per entrambi gli scopi nel progetto è la *InceptionV3*, un'architettura di rete neurale convoluzionale ottimizzata nell'efficienza dei parametri proposta da [Szegedy et al. 2015]. La scelta è ricaduta su questo modello perchè, a differenza di altri modelli altrettanto efficaci ed efficienti, sono pubblicamente disponibili modelli già allenati su dataset di annotazione generici molto completi (e.g. OpenImages[2]) che comprendono molti dei concetti che sono di interesse per SmartNews. Per una descrizione più approfondita dell'architettura scelta, si prega di fare riferimento alla Sezione 3.

[2] https://github.com/openimages/dataset

2.2 Indicizzazione e Ricerca

Alla base del paradigma di ricerca per immagini c'è l'idea di confrontare le feature estratte dall'immagine di query con le feature estratte delle immagini del dataset su cui si vuole effettuare la ricerca. Opportune misure di distanza o dissimilarità vengono utilizzate per il confronto dei descrittori numerici usati per la rappresentazione del contenuto visuale delle immagini. Tipicamente, il processo di confronto tra i descrittori della query e quelli dell'immagine del dataset non viene eseguito in maniera sequenziale poiché nella maggior parte dei casi il dataset da interrogare è troppo grande o la misura usata per il confronto è troppo costosa per poter permettere di effettuare la ricerca in tempi ragionevoli. I descrittori estratti dalle immagini devono essere quindi organizzati in indici di ricerca, come ad esempio gli indici metrici [Zezula et al. 2006]. Tuttavia molte delle feature visuali sono rappresentate da descrittori numerici di grandi dimensioni. Per esempio, le deep feature estratte usando la popolare architettura AlexNet [Krizhevsky et al. 2012] hanno migliaia di dimensioni (4,096 dimensioni se si considera l'output del penultimo livello della rete o addirittura 9,216 dimensioni per l'output dell'ultimo livello convoluzionale). Questo ha rappresentato un primo ostacolo per l'uso delle deep feature su larga scala, a causa del cosiddetto fenomeno del "*curse of dimensionality*" per cui molti degli indici di ricerca su feature di grandi dimensioni non hanno performance migliori rispetto alla semplice ricerca sequenziale. Un approccio efficace per affrontare questa problematica è l'uso di indici di ricerca approssimata, come le tecniche di *Local Sensitive Hashing* [Indyk & Motwani 1998, Bawa et al. 2005, Lv et al. 2007] ed gli approcci di *Permutation-Based Indexing* (PBI) [Amato et al. 2014, Chavez et al. 2008, Novak et al. 2011].

Nell'ambito del progetto SmartNews sono stati investigati e utilizzati metodi di tipo PBI, in quanto essi hanno mostrato ottimi risultati per la ricerca di deep feature in database visuali di enormi dimensioni [Amato et al. 2017, Novak & Zezula 2016].

L'idea cardine dei metodi basati sulle permutazioni è quella di trasformare ciascun oggetto (nel nostro caso il vettore numerico associato alla feature di un'immagine) con una sequenza di identificatori (permutazione) in modo che oggetti simili abbiano permutazioni simili. La ricerca viene eseguita individuando tutti quegli oggetti la cui permutazione è più simile alla permutazione dell'oggetto di query.
Tipicamente, la permutazione di un oggetto viene calcolata riordinando gli identificatori di un insieme di oggetti di riferimento (*pivots*) in base alla loro distanza dall'oggetto da rappresentare. Un esempio di calcolo della permutazione di un oggetto (o_1) dati quatro pivots (p_1, p_2, p_3, p_4) è illustrata in Figura 4. L'intuizione alla base di questo tipo di rappresentazione è che oggetti simili avranno una similare visione del mondo che li circonda (comprese le distanze relative dai pivots), per cui le permutazioni ad essi associati saranno simili.

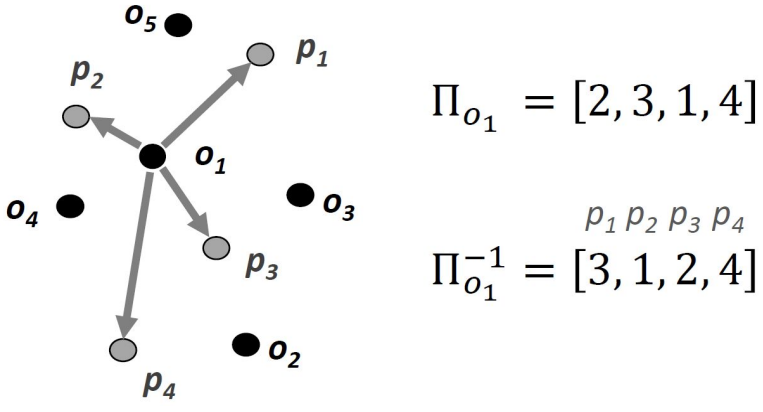


Figura 4: Esempio di permutazione (Π_{o_1}) associata ad un oggetto oggetto o_1 dati quatro pivots (p_1, p_2, p_3, p_4). In questo caso Π_{o_1}=[2,3,1,4] poiché il pivot più vicino all'oggetto considerato è p_2, il secondo pivot più vicino è p_3, seguito da p_1 e p_4. In figura è riportata anche una rappresentazione equivalente, detta permutazione inversa, indicata con $\Pi_{o_1}^{-1}$, che viene spesso utilizzata nella pratica per la comparazione degli oggetti.

In SmartNews è stata svolta un'attività di ricerca per rappresentare le deep feature in maniera efficiente ed efficace mediante l'uso delle permutazioni. In particolare è stata sviluppata una tecnica, detta *Deep Permutations*, che utilizza i valori di attivazione del layer intermedio di una rete (ossia le singole componenti di una deep feature) per la generazione di una permutazione senza il calcolo delle distanze tra la deep feature ed un insieme di pivot. In questo caso, gli indici usati nella permutazione sono gli indici delle coordinate della deep feature. Tali elementi vengono riordinati in maniera decrescente. L'ordinamento indotto negli indici delle coordinate della deep feature è quello che fornisce la permutazione risultante. Un esempio di calcolo di una Deep Permutation e della sua rappresentazione inversa è raffigurato in Figura 5.

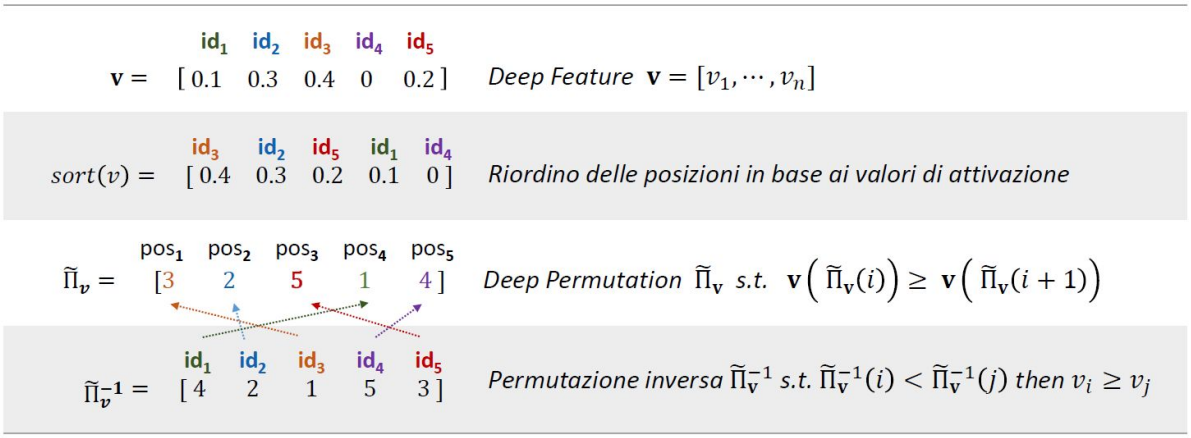


Figura 5: Esempio di calcolo della Deep Permutation $\overline{\Pi}_v$ *associata ad una deep feature* v *.*

L'intuizione alla base delle Deep Permutation è che le deep feature racchiudono delle informazioni di alto livello che possono essere utilizzate per la generazione della permutazione. Si può pensare, ad esempio, che ciascuna dimensione del vettore di una deep feature rappresenti una certo concetto visuale, e che il valore in tale dimensione specifichi l'importanza di tale concetto all'interno dell'immagine considerata. Riordinare le dimensioni di una deep feature in base ai valori delle componenti può essere interpretato come rappresentare una immagine in base ai concetti visuali pù rilevanti all'interno delle immagini.

Per i dettagli sulla sperimentazione fatta ed risultati ottenuti mediante l'uso delle Deep Permutation si rimanda alla seguente pubblicazione scientifica:

> Amato, G., Falchi, F., Gennaro, C., & Vadicamo, L. (2016, October). Deep permutations: deep convolutional neural networks and permutation-based indexing. In *International Conference on Similarity Search and Applications* (pp. 93-106). Springer International Publishing.

E' importante osservare che la tecnica proposta in SmartNews non solo ha permesso di ridurre il costo computazionale per il calcolo delle permutazioni, ma ha mostrato anche una migliore efficacia rispetto all'approccio basato sulle permutazioni tradizionali [Amato et al. 2016b].

Inoltre, per il confronto e la sperimentazione di varie tecniche di ricerca per similarità in archivi di deep feature, in SmartNews è stato creato un importante benchmark, chiamato YFCC100M-HNfc6, per la valutazione della ricerca per similarità. Esso contiene tre tipi di deep feature estratte da 97 milioni di immagini ed i risultati precalcolati della ricerca k-NN (*k-Nearest Neighbors*) per 1000 query e k=10,000. Tale benchmark è stato presentato e descritto nelle seguenti pubblicazioni a conferenza:

- Amato, G., Falchi, F., Gennaro, C., & Rabitti, F. (2016, October). YFCC100M-HNfc6: a large-scale deep features benchmark for similarity search. In *International Conference on Similarity Search and Applications* (pp. 196-209). Springer, Cham.

- Amato, G., Falchi, F., Gennaro, C., & Rabitti, F. (2016, October). YFCC100M HybridNet fc6 Deep Features for Content-Based Image Retrieval. In *Proceedings of the 2016 ACM Workshop on Multimedia COMMONS* (pp. 11-18). ACM.

Dal punto di vista implementativo, la tecnica delle Deep Permutation è facilmente realizzabile con indici testuali pre-esistenti grazie alla tecnica di *Surrogate Text Representation* (STR) [Gennaro et al. 2010]. Nella Surrogate Text Representation, una permutazione è rappresentata all'interno di un indice testuale come un documento contenente un testo surrogato. L'indicizzazione e la successiva ricerca di questo testo surrogato mediante indici testuali che implementano il *vector space model* [Manning et al. 2008] (e.g. Lucene) permette di imitare le operazioni di ricerca nello spazio delle permutazioni, ottenendo gli stessi risultati senza implementare appositamente un indice per permutazioni.

Nella STR, l'ordine di un permutante (che ne determina l'importanza) viene codificato ripetendo una parola a lui associata. Più il rank del permutante è alto, più volte la parola viene ripetuta. In questo modo, nell'indice testuale opportunamente configurato, questa parola avrà un peso maggiore nel recupero del documento associato alla permutazione.

Nell'ambito del progetto, è stata studiata l'applicazione della tecnica *Deep Permutations* e *Surrogate Text Representation* a rappresentazioni di immagini estratte da reti neurali (*deep features*). In particolare, è stato effettuato uno studio sull'efficienza dell'indicizzazione e ricerca di immagini con features R-MAC (precedentemente introdotte nella Sezione 2.1), dato che queste sono lo stato dell'arte nell'instance image retrieval con deep features convoluzionali. Sono state implementate e ottimizzate le trasformazioni matematiche che permettono di generare una permutazione dai valori delle features visuali, ed è stata definita una funzione di similarità tra queste ultime basata sul prodotto scalare. Sono stati condotti esperimenti per analizzare il trade-off tra efficacia (in termini di qualità dei risultati) ed efficienza (in termini di tempo di query) che nasce dall'approssimazione delle permutazioni tramite una loro troncatura ai soli primi K elementi. I risultati di tale attività di ricerca sono descritti nella seguente pubblicazione alla conferenza internazionale ICMR-2017:

> Amato G., Carrara F., Falchi F., Gennaro C., Efficient Indexing of Regional Maximum Activations of Convolutions using Full-Text Search Engines. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval* (pp. 420-423). ACM.

Come già accennato in Sezione 2.1, successivamente si sono selezionate *deep features* diverse da R-MAC per la rappresentazione di immagini in SmartNews, ma data la generalità dell'approccio sviluppato, non si sono riscontrati problemi nell'applicazione dello stesso metodo di indicizzazione alle deep features estratte nel contesto del progetto. Il modulo di indicizzazione e ricerca è stato quindi implementato mediante l'applicazione di Deep Permutations e Surrogate Text Representation alle deep features estratte dalle immagini con l'annotatore InceptionV3. Il testo così generato è stato indicizzato con Lucene.

3 Annotazione automatica

L'annotazione automatica delle immagini consiste nell'assegnare automaticamente alle immagini delle etichette semantiche come parole, frasi o periodi che descrivono gli oggetti, gli attributi, le azioni e le scene rappresentati in un'immagine. Lo scopo finale dell'annotazione è generalmente rendere ricercabili, tramite etichette, testo libero e attributi, le immagini stesse. La ricerca in questo ambito è stata molto attiva negli ultimi anni, in particolare i settori del Multimedia Information Retrieval della Computer Vision. L'uso di tecnologie di intelligenza artificiale basate su reti neuronali e il deep learning è stato non solo fondamentale per lo sviluppo più recente di questo settore, ma ha segnato anche il punto di rinascita degli studi sull'intelligenza artificiale a partire dal 2012.

Un momento fondamentale per lo sviluppo delle tecnologie di annotazione automatica delle immagini è stato il rilascio nel 2008 del dataset MIRFLICKR, una collezione di 1 milione di immagini annotate manualmente. Grazie anche al lavoro svolto all'interno di ImageCLEF negli anni successivi sono stati sviluppati algoritmi per l'assegnazione

automatica alle immagini di etichette indicanti elementi (nebbia, riflessi, etc.), caratteristiche della scene (costa, skyline, etc.) e emozioni (euforia, paura, etc.). Di fondamentale importanza è stato successivamente il dataset ImageNet usato nelle significative competizioni di Large Scale Visual Recognition (ILSVRC). ImageNet è un dataset di immagini ognuna delle quali è assegnata a un synset di WordNet e che include oggi più di 14 milioni di immagini associati a più 20,000 synset.

Ancor più recentemente il dataset e la competizione Microsoft COCO hanno permesso significativi miglioramenti in campi più specifici quali il captioning, cioè la descrizione automatica delle immagini tramite frasi complesse di senso compiuto.

Fino al 2012 gli approcci migliori, come evidenziano i risultati nell'ambito delle varie competizioni internazionali (ILSVRC e ImageCLEF su tutte), erano basati su una combinazione di feature globali, come l'istogramma dei colori, e locali (SIFT, SURF etc…) e classificatori SVM o logistic regression. Le competizioni legate a Microsoft COCO, più recente, sono invece sempre state caratterizzate da un uso intensivo di metodi di deep learning.

Da un punto di vista del software liberamente disponibile in rete, di eccezionale importanza è stato il framework per deep learning denominato Caffe (Convolutional Architecture for Fast Feature Embedding), sviluppato principalmente dall'università di Berkeley, accompagnato da un punto di raccolta, il Caffe Model Zoo[3] in cui è possibile trovare moltissime reti neurali già allenate e usabili liberamente che permettono l'annotazione automatica di immagini e molti altri task legati alla computer vision.

Nell'ambito del progetto ci si è concentrati su un dataset e una rete convoluzionale ancora più recenti e il cui sviluppo è legato principalmente a Google. The OpenImages Dataset[4] contiene 9 milioni di immagini annotate con label di alto livello e bounding boxes associate a circa 20,000 classi. Il dataset contiene un misto di machine generated labels e human verified labels. Nel set di label presenti in OpenImages ci sono anche quelle che nella proposta del progetto erano "tipologia di immagine" (gruppi di persone, ritratto, etc…).

Attraverso l'uso di questo dataset per il training è stato possibile realizzare una rete per l'annotazione automatica delle immagini, usata nel progetto, basata sull'architettura convoluzionale Inception V3

Come esempio di risultato ottenibile con questa rete si veda la seguente immagine:

```
3473: /m/04rky - mammal (score = 0.89)
3981: /m/068hy - pet (score = 0.87)
1261: /m/01yrx - cat (score = 0.87)
5723: /m/0jbk - animal (score = 0.83)
4605: /m/09686 - vertebrate (score = 0.81)
841: /m/01l7qd - whiskers (score = 0.68)
2430: /m/0307l - cat-like mammal (score = 0.68)
4349: /m/07k6w8 - small to medium-sized cats (score = 0.67)
5643: /m/0hjzp - kitten (score = 0.30)
50: /m/012c9l - domestic short-haired cat (score = 0.17)
```

L'architettura di rete neurale che abbiamo usato in specifico è la Inception V3 [Szegedy et al. 2015]. Questa architettura è un'evoluzione della GoogleNet precedentemente proposta dagli stessi autori. L'idea principale di questa architettura si basa sull'utilizzo di un blocco computazionale chiamato *Inception*, che permette di estrarre informazione a più risoluzioni ad ogni livello della rete, applicando al suo input più convoluzioni di diverse dimensioni e concatenando i relativi output in un'unica rappresentazione. La terza iterazione (V3) di questa architettura apporta dei miglioramenti all'efficienza della rete, in particolare agendo sulla composizione del blocco Inception. C'è stato uno sforzo da parte degli autori nel ridurre il carico computazionale delle convoluzioni con finestre grandi (e.g 5x5, 7x7) sostituendo queste ultime con delle versioni fattorizzate, composte da una cascata di convoluzioni più piccole (e.g. 3x3 + 3x3, oppure 5x1 + 1x5; vd. Figura 6).

[3] https://github.com/BVLC/caffe/wiki/Model-Zoo
[4] https://github.com/openimages/dataset

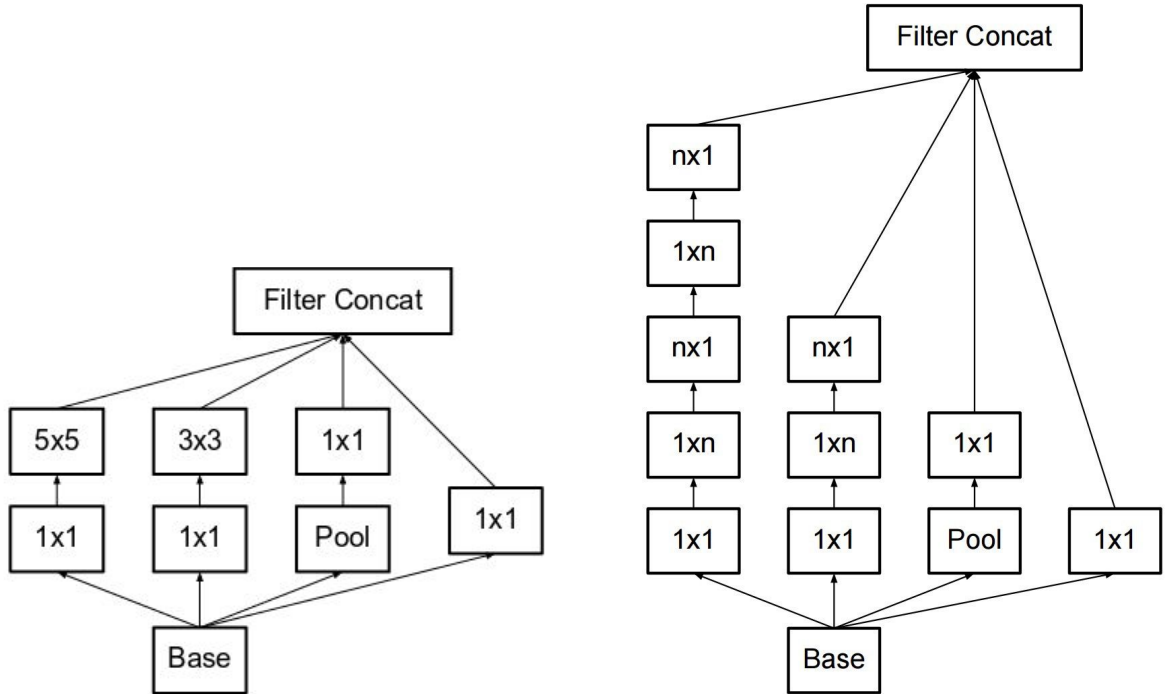


Figura 6: Composizione di un blocco Inception standard (sulla sinistra) e la sua versione ottimizzata con convoluzioni separabili (sulla destra).

Il modello risultante ha raggiunto performance allo stato dell'arte nella competizione ILSVRC12 sulla classificazione di immagini su larga scala con un errore top-5 di circa 4% e con un costo computazionale relativamente basso rispetto a modelli concorrenti (come VGGNet).

Nell'ambito del progetto sono stati considerati anche approcci diversi i cui risultati scientifici preliminari, incoraggianti, sono riportati nel paper [Amato et al. 2017] che si basa su un recente dataset di 100 milioni di immagini provenienti da Flickr (YFCC100M[5]). Il dataset, fortemente rumoroso perché contenente immagini poco annotate e spesso annotate in modo errato, è oggetto di molte ricerche fra cui quelle sul CBIR [Amato et al. 2016c].

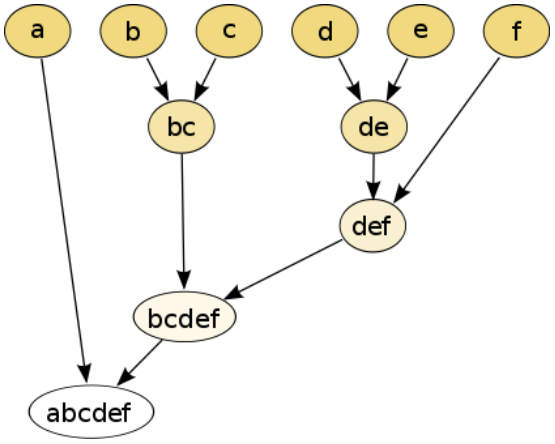
Un settore recente di interesse per il progetto è infine quello del cross-media retrieval che permette la ricerca tramite testo di immagini senza che queste siano state annotate automaticamente o manualmente. Questo è stato possibile attraverso la trasposizione del testo inserito come query dall'utente in feature visuale che poi viene usata per ricercare le immagini più rilevanti per il testo dato attraverso l'uso di sistemi di CBIR. I risultati di questi studi sono riportati nel paper [Carrara et al. 2017].

4 Clustering e near duplicates

Come previsto dalla proposta di progetto, è stato implementato all'interno dell'attività 3.3 "Visual Content Mining" un sistema di clusterizzazione e rilevamento di duplicati per gruppi di immagini. Lo scopo di questa attività è quella di raggruppare immagini in gruppi visualmente e semanticamente coerenti per molteplici fini, quali l'analisi multimediale di particolari gruppi o sorgenti social, l'individuazione di immagini uniche, duplicate o quasi-duplicate e l'aiuto alla costruzione di sistemi di visualizzazione e navigazione di immagini.

[5] https://multimediacommons.wordpress.com/yfcc100m-core-dataset/

Data la molteplicità dei fini al quale lo specifico algoritmo di clustering deve soddisfare, si è optato per un algoritmo di clustering gerarchico agglomerativo (o "bottom-up") [Ward 1963]. In questo tipo di algoritmo, ogni elemento viene posto inizialmente in un proprio cluster e ad ogni iterazione un nuovo cluster viene ottenuto unendo i due cluster della precedente iterazione. L'output che si ottiene è un albero binario che rappresenta la gerarchia dei cluster formati ad ogni iterazione. Un esempio di gerarchia prodotta è la seguente:



Questa struttura gerarchica di aggregazione presenta i seguenti vantaggi:

- flessibilità nella scelta della granularità di aggregazione, in quanto questa può essere scelta decidendo a quale altezza tagliare l'albero binario. In SmartNews, questo requisito è fondamentale ai fini della visualizzazione, in quanto permette di disaccoppiare l'algoritmo di clustering da eventuali parametri dell'algoritmo di visualizzazione, come il numero minimo o massimo di gruppi e/o sottogruppi di elementi.
- facile individuazione di outliers, in quanto questi si posizionano nella gerarchia più vicini alla radice dell'albero. In SmartNews, l'individuazione di outliers è fondamentale per scoprire comportamenti e contenuti unici all'interno di una raccolta multimediale.
- individuazione di immagini duplicate o quasi-duplicate implicita nell'algoritmo, in quanto i primi nodi formati dall'algoritmo (i più vicini alle foglie della gerarchia) raccolgono questo tipo di duplicati. Una semplice analisi della distanza media tra gli elementi di questi nodi permette facilmente di individuare un gruppo di duplicati o quasi-duplicati.

Per l'implementazione in SmartNews, abbiamo utilizzato la libreria *scikit-learn*[6] per Python, open source con licenza BSD.

Come rappresentazione delle immagini sul quale il clustering è svolto, abbiamo optato per le deep features estratte dal classificatore OpenImages. Dato che i risultati del clustering hanno come requisito di raggruppare immagini con simile semantica dal punto di vista umano, si è scelto di usare uno strato intermedio di una rete allenata per task di annotazione, perché più vicina alla percezione semantica, scartando features per instance image retrieval (come RMAC) più adatte a descrivere i dettagli a basso livello dell'immagine.

[6] http://scikit-learn.org/stable/modules/clustering.html#hierarchical-clustering

5 Analisi del sentimento

Anche se non previsto dalla Scheda Tecnica del progetto, a completamento dell'attività 3.3 "Visual Content Mining" è stato intrapreso lo studio e lo sviluppo di una tecnica innovativa per l'analisi del sentimento delle immagini allo scopo di fornire all'utente un ulteriore ed innovativo strumento di analisi dei dati visuali. La scelta di affrontare la tematica dell'analisi del sentimento delle immagini è stata suggerita da notevoli fattori quali la rilevanza scientifica di questo tipo di analisi, la cospicua quantità di dati multimediali resi disponibili nell'ambito del progetto e la disponibilità di un classificatore del sentimento per il testo realizzato nell'Attività 3.2 "Text Mining".

Negli ultimi anni, la ricerca scientifica ha fatto notevoli progressi nel campo dell'analisi del sentimento. Tuttavia, questa tematica di ricerca è stata affrontata principalmente nel settore dell'analisi testuale, dove il problema dell'individuazione automatica del sentimento ha originariamente attratto l'attenzione di numerosi ricercatori sia nell'ambito accademico che industriale. Recentemente, il problema dell'individuazione del sentimento di dati visuali ha acquisito un forte e crescente interesse, alimentato soprattutto dalla diffusa mancanza di una descrizione testuale di tali dati. Si pensi, per esempio, al trend seguito dagli utenti dei social network che sono abituati a scattare e condividere le proprie foto senza descriverne propriamente il contenuto.

Se da un lato, le immagini sono un efficace canale di comunicazione che permette di trascendere barriere linguistiche e culturali, ossia "un'immagine vale più di mille parole" come recita un antico detto, dall'altro le emozioni ed i sentimenti che esse inducono nell'osservatore sono molto soggettive. Tale soggettività unita al cosiddetto *affettive gap* [Siersdorfer et al. 2010a], vale a dire il divario tra il reale contenuto sentimentale delle immagini ed i descrittori numerici usati per la loro rappresentazione, rende l'analisi del sentimento delle immagini un problema complesso da affrontare. Questi fattori riflettono anche la difficoltà nel reperire dataset che contengano molte immagini annotate accuratamente e che quindi possano essere usate per allenare dei classificatori visuali per il sentimento.

In SmartNews, l'attività svolta in questo contesto si è incentrata nella costruzione di un classificatore visuale per l'individuazione della polarità (positiva, negativa o neutra) del sentimento delle immagini.

I risultati di questa attività sono stati presentati nella pubblicazione scientifica:

> "Vadicamo, L., Carrara, F., Cimino, A., Cresci, S., Dell'Orletta, F., Falchi, F., & Tesconi, M. (2017). *Cross-Media Learning for Image Sentiment Analysis in the Wild*. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 308-317)"

Un aspetto innovativo della nostra attività è stato quello di affrontare questa tematica di ricerca usando dei dati non controllati, ossia seguendo il trend dell'analisi di un problema "in the wild" che prevede l'uso di dati non etichettati da annotatori umani e non selezionati dal web usando particolari parole chiave di ricerca relative al sentimento. A questo scopo è stata selezionata una grande quantità di dati multimediali dai social media; la selezione è stata fatta in maniera casuale così da avere una vasta variazione nei contenuti dei dati utilizzati.

Si faccia presente che ad oggi, la maggior parte dei classificatori visuali proposti in letteratura sono stati addestrati su dataset annotati sentimentalmente tramite servizi di crowd-sourcing, od utilizzando immagini collezionate dal web mediante termini di ricerca relativi al sentimento. A differenza di questi approcci supervisionati, in SmartNews abbiamo proposto una procedura "self-supervised" a partire da milioni di tweet (testo ed immagini) non etichettati. I dati sono stati raccolti da IIT-CNR mediante il Twitter Stream Crawler da essi sviluppato. Tali dati sono stati etichettati automaticamente analizzandone il testo mediante un classificatore testuale sviluppato da ILC-CNR basato su tecniche di deep learning. Infine l'annotazione testuale è stata utilizzata per etichettare le immagini da utilizzare per l'allenamento del classificatore visuale, basato anch'esso su le recenti tecniche di deep learning. Maggiori dettagli sono forniti nella Sezione 5.2.

Nonostante l'annotazione testuale dei tweet può non rispecchiare il contenuto visuale delle immagini associate, come nel caso di commenti ironici o non rilevanti, è stato dimostrato sperimentalmente che il nostro approccio permette di

allenare un efficace classificatore visuale del sentimento. Infatti, i nostri risultati su benchmark di immagini annotate a mano hanno mostrato che l'accuratezza della predizione del nostro classificatore visuale supera le prestazioni di altri classificatori proposti in letteratura, nonostante quest'ultimi siano allenati in maniera supervisionata su dati "controllati" e relativi al sentimento (si veda anche la Sezione 5.2).

Un importante output di questa attività è stata l'analisi di più di 3 milioni di tweet che ha portato alla creazione del *Twitter for Sentiment Analysis* (T4SA) dataset, composto da circa 1 milione di tweet annotati testualmente con elevata confidenza e dalle corrispondenti 1.4 milioni di immagini (ciascun tweet può essere associato a più immagini). Tale dataset è stato utilizzato per l'allenamento del classificatore visuale. Sia il dataset che i classificatori addestrati su di esso sono stati resi disponibili pubblicamente all'indirizzo http://www.t4sa.it/ .

Di seguito, nella Sezione 5.1 presentiamo brevemente i più recenti e rilevanti lavori scientifici sull'analisi visuale del sentimento; nella Sezione 5.2 descriviamo l'approccio proposto in Smart News ed i principali risultati ottenuti.

5.1 Stato dell'arte sull'analisi visuale del sentimento

Nonostante esista un'estesa letteratura scientifica sull'analisi del sentimento di un testo, la ricerca sull'analisi del sentimento di una immagine è ancora in una fase iniziale. I primi approcci per la predizione del sentimento di una immagine sono stati basati su approcci supervisionati o semi-supervisionati che avevano lo scopo di mappare descrittori a basso livello delle immagini in emozioni umane [Machajdik & Hanbury 2010, Siersdorfer et al. 2010b] . Per superare l'*affective gap* tra le feature usate per rappresentare le immagini ed il loro contenuto affettivo, Borth et al. (2013), Yuan et al. (2013) e Jou et al. (2015) hanno proposto l'uso di entità visuali ed attributi di più alto livello per la rappresentazione delle immagini. Il loro principale contributo è stata la creazione di due estese ontologie del sentimento visuale, chiamate Visual sentiment Ontology (VSM) e Multilingual-VSM, che consistono in migliaia di oggetti semantici (coppie aggettivi-nomi, dette ANP) legate alle emozioni. Usando queste ontologie essi hanno proposto anche dei detector (*Sentibank* e *MVSO*) per estrarre dalle immagini delle rappresentazioni del sentimento descritte mediante le ANP. Recentemente Li et al. (2017) ha arricchito tale rappresentazione fondendo la predizione del sentimento dedotto dalle feature di immagini con la predizione testuale delle ANP associate all'immagine. Gli approcci usati in [Jou et al 2015, Li et al 2017], si basano su tecniche di deep learning che recentemente hanno portato all'avanzamento dello stato dell'arte in numerosi domini di ricerca. Seguendo il successo del deep learning, molti altri approcci basati su reti neurali "deep" sono stati proposti per l'analisi del sentimento visuale [Campos et al. 2017, Chen et al. 2014, Islam & Zhang 2016, Rao et al. 2016, You et al. 2015]. La maggiorparte di questi approcci hanno in comune l'uso di una Convolutional Neural Network (CNN) addestrata o riadattata (fine-tuning) su immagini annotate con etichette relative al sentimento. A tale scopo, in [Campos et al. 2017, Chen et al. 2014, Islam & Zhang 2016, Jou et al 2015] sono state utilizzate le più comuni architetture CNN (come AlexNet, PlaceCNN e GoogleNet), mentre altri autori come [You et al. 2015] hanno utilizzato una architettura CNN appositamente disegnata per la predizione del sentimento.

Vi sono anche numerosi lavori che hanno affrontato l'analisi del sentimento utilizzando più modalità, come testo ed immagini. Per esempio Cao et al. (2016) ha proposto una fusione delle predizioni del sentimento ottenute dalla singole analisi del testo e delle immagini, metre You et al. (2016) ha proposto un sistema di "cross-modality", detto CCR, per predirre il sentimento analizzando contestualmente testo ed immagini.

A differenza degli approcci sopra citati, in SmartNews abbiamo allenato un classificatore visuale senza alcuna conoscenza a priori dei dati utilizzati per l'addestramento della CNN, affrontando il complesso problema dell'analisi del sentimento a partire da una grande mole di dati raccolti dai social media. Un lavoro in letteratura che va in questa direzione è [Wand et al. 2015] dove è stato proposto un approccio non supervisionato per l'analisi del sentimento a partire da dati multimediali. Tuttavia esso utilizza congiuntamente testo ed immagini per la predizione del sentimento, mentre la rete neurale addestrata in SmartNews permette l'analisi del sentimento delle immagini anche se non accompagnate da testo.

5.2 Analisi visuale del sentimento proposta in Smart News

Il classificatore visuale per il sentimento proposto in Smart News è stato realizzato mediante

1. la raccolta di una grande quantità di dati da Twitter contenenti sia sia immagini che testo;
2. l'analisi testuale del sentimento dei tweet raccolti;

3. la selezione dei tweet classificati con elevata accuratezza;
4. la costruzione del *Twitter for Sentiment Analysis* (T4SA) dataset, contenente le immagini annotate automaticamente in base alla classificazione testuale del sentimento dei tweet associati;
5. utilizzo di una partizione bilanciata del T4SA dataset per l'allenamento del classificatore visuale.

L'attività di raccolta dati è stata svolta dal partner IIT-CNR. I dati (sia testo che immagini) sono stati selezionati da Twitter, sfruttando il "Twitter's Sample API"[7] che permette di accedere ad un campione random dell'1% dello stream globale di tutti i tutti i prodotti. La raccolta dati è durata 6 mesi, a partire dal mese di Luglio 2016. Al fine di mantenere solo i dati utili per la costruzione del T4SA dataset, tutti i dati raccolti sono stati filtrati seguendo quattro semplici regole che prevedevano l'esclusione di:

A. retweet;
B. tweet che non contenevano nessuna immagine statica;
C. tweet il cui testo non era scritto in lingua Inglese;
D. tweet il cui testo era più corto di cinque parole.

Le Regole A e B sono servite per aumentare la qualità dei dati raccolti. In particolare la Regola A ha avuto lo scopo di evitare la ridondanza dei dati collezionati, evitando la raccolta di un grande numero di immagini duplicate. La Regola B ha garantito che i tweet raccolti contenessero almeno un'immagine utilizzabile per la costruzione del T4SA dataset (da usare per il successivo allenamento di un classificatore visuale). Le Regole C e D, invece, hanno avuto lo scopo di garantire che i testi raccolti fossero adatti alla classificazione testuale del sentimento.

Le precedenti regole hanno portato al filtraggio di circa il 98.7% di tutti i dati collezionati dallo stream di Twitter. In ogni caso, l'enorme volume di tweet prodotti globalmente ha permesso la collezione di circa 43 tweet utili (ossia non filtrati) al minuto, per un totale di circa 3.4 milioni di tweet corrispondenti a circa 4 milioni di immagini.

I testi dei tweet raccolti sono stati classificati da ILC-CNR mediante l'analisi testuale del sentimento. A tale scopo l'ILC ha sviluppato un classificatore testuale del sentimento adattando l'"ItaliaNLP Sentiment Polarity Classifier " [Cimino & Dell'Orletta 2016] alla lingua inglese. Il metodo utilizzato combina una rete neurale ricorrente di tipo Long Short Term Memory (LSTM) (per l'estrazione delle feature e l'individuazione di dipendenze temporali tra i termini) ed una Support Vector Machine (SVM) (per la classificazione). Come riportato in [Cimino & Dell'Orletta 2016]: "Le SVM combinano la rappresentazione del documento prodotta da LSTM con un ampio insieme di features che descrivono la struttura lessicale e grammaticale del testo".

Il classificatore testuale del sentimento è stato utilizzato per analisi dei testi dei 3.4 milioni di tweet raccolti. I tweet i cui testi sono stati classificati con una confidenza alta (maggiore dello 0.85) sono stati selezionati per formare un dataset annotato di immagini, dove ciascuna immagine è stata etichettata in base al sentimento del testo ad essa associato. Questa prima selezione ha portato ad un primo insieme di dati costituito da circa un milione di tweet, di cui precisamente 371,341 etichettate come Positive, 629,566 etichettate come Neutre e solo 31,327 etichette come Negative. Come spesso accade nell'ambito dell'analisi del sentimento nei social media [Li et al. 2011], il dataset ottenuto è risultato molto sbilanciato. Al fine di incrementare il numero di immagini etichettate come Negative, è stato impostato una soglia di accuratezza più bassa per la selezione di questa classe, per un totale di 179,395 tweet Negativi.[8] Il dataset risultante, denominato *Twitter for Sentiment Analysis* (T4SA), contiene un totale di 1,473,394 immagini relative a 1,179,957 tweet, dove ciascuna immagine è stata annotata in base alla polarità del testo corrispondente.

Al fine di individuare un sottoinsieme di T4SA utile per l'apprendimento di un classificatore visuale, sono state eliminate le immagini corrotte o duplicate, ed è stata selezionato un partizionamento bilanciato contenente 156,862 immagini per ciascuna classe.

Le composizioni finali del T4SA e del suo partizionamento bilanciato sono riportate nella Tabella 1.

[7] https://dev.twitter.com/streaming/reference/get/statuses/sample

[8] Nei nostri esperimenti la differenza nella precisione della classificazione di immagini in positiva, negativa o neutra non ha mai superato l'1% e quindi l'uso di una soglia più bassa per la selezione dei testi negativi non ha dato nessun impatto sulla qualità del classificatore visuale finale.

Sentimento	T4SA (tweets)	T4SA (immagini)	T4SA (no near duplicates) (immagini)	B-T4SA (immagini)
Positivo	371,341	501,037	372,904	156,862
Neutro	629,566	757,895	444,287	156,862
Negativo	179,050	214,462	156,862	156,862
Tot	1,179,957	1,473,394	974,053	470,586

Tabella 1: T4SA Dataset e suoi sottoinsiemi.

Il dataset B-T4SA è stato utilizzato per fare il fine-tuning di una deep CNN, così da ottenere un classificatore visuale del sentimento. Infatti, dato il grande numero di parametri da determinare, per allenare da zero una CNN sarebbero state necessarie milioni di immagini etichettate. Quando, come nel nostro caso, non si ha a disposizione una così grande mole di dati etichettati, è possibile far leva su tecniche di transfer learning come il fine-tuning. In questo caso, i parametri o alcuni dei parametri di una rete pre-allenata vengono riallenati per un task diverso ma correlato a quello usato per allenare la rete originaria. Per costruire il classificatore visuale di SmartNews, abbiamo testato i classificatori ottenuti facendo il fine-tuning di due reti pre-allenate per l'annotazione semantica: la AlexNet [Zhou et al. 2014] e la VGG-19 [Simonyan & Zisserman 2014]. Ulteriori dettagli tecnici e gli esperimenti svolti sono riportati in [Vadicamo et al. 2017]. Il classificatore ottenuto a partire dalla VGG-19 ha dato i migliori risultati sia sul test di B-T4SA che sul benchmark Twitter Testing Dataset [You et al 2015] che contiene dati annotati annotati a mano. Inoltre, il nostro classificatore visuale basato sulla VGG-19 ha mostrato performance migliori degli altri classificatori visuali proposti in letteratura: si veda, ad esempio, la Figura 7 estratta dalla pubblicazione [Vadicamo et al. 2017].

E' importante osservare che il B-T4SA usato per l'allenamento della rete, non essendo annotato a mano, può contenere annotazioni errate specialmente quando il testo del tweet non rispecchia il contenuto delle immagini ad esso associate. Tuttavia l'aver usato un grande quantitativo di dati ha permesso di allenare il classificatore anche in presenza di dati rumorosi. In Figura 8 riportiamo un esempio delle immagini classificate con maggiore confidenza. E' interessante osservare che, da un punto di vista qualitativo, in molti casi la predizione del classificatore visuale risulta essere migliore di quella inferita dall'analisi del sentimento del testo associato all'immagine. Inoltre la predizione della classe Neutro risulta molto utile per identificare immagini di oggetti e prodotti commerciali.

Method	Training details	Twitter Testing Dataset		
		5 agree	≥ 4 agree	≥ 3 agree
Approaches without intermediate fine-tuning				
GCH [42] (res from [50]) *	-	0.684	0.665	0.66
SentiBank [8] (res from [50]) °	-	0.709	0.675	0.662
LCH [42] (res from [50]) *	-	0.710	0.671	0.664
GCH+ BoW [42] (res from [50]) *	-	0.710	0.685	0.665
LCH+ BoW [42] (res from [50]) *	-	0.717	0.697	0.664
Sentribute [51] (res from [50]) °	-	0.738	0.709	0.696
CNN [50] •	Custom architecture *tr* on Flickr (VSO) [8]	0.783	0.755	0.715
AlexNet [23] (res from [9]) •	AlexNet [23] *tr* on ILSVRC2012 [40]	0.817	0.782	0.739
PlaceCNN [52] (res from [9]) •	AlexNet [23] *tr* on Places205 [52]	0.830	-	-
GoogleNet [44] (res from [20]) •	GoogleNet [44] *tr* on ILSVRC2012 [40]	0.861	0.807	0.787
HybridNet •	AlexNet [23] *tr* on (ILSVRC2012 [40] + Places205 [52])	0.867	0.814	0.781
VGG-19 •	VGG-19 [43] *tr* on ILSVRC2012 [40]	0.881	0.835	0.800
Approaches using an intermediate fine-tuning				
PCNN [50] •	Custom architecture *tr* on Flickr (VSO) [8] + *ft* on Flickr (VSO) [8]	0.773	0.759	0.723
DeepSentiBank [11] (res from [9]) °•	AlexNet [23] *tr* on ILSVRC2012 [40] + *ft* on Flickr (VSO) [8]	0.804	-	-
MVSO [EN] [22] (res from [9]) °•	DeepSentiBank [11] *ft* on MVSO-EN [22]	0.839	-	-
Hybrid-T4SA FT-A (Ours) •	AlexNet [23] *tr* on (ILSVRC2012 [40] + Places205 [52]) + *ft* on B-T4SA	0.864	0.830	0.800
Hybrid-T4SA FT-F (Ours) •	AlexNet [23] *tr* on (ILSVRC2012 [40] + Places205 [52]) + *ft* on B-T4SA	0.873	0.832	0.810
VGG-T4SA FT-F (Ours) •	VGG-19 [43] *tr* on ILSVRC2012 [40] + *ft* on B-T4SA	0.889	0.857	0.815
VGG-T4SA FT-A (Ours) •	VGG-19 [43] *tr* on ILSVRC2012 [40] + *ft* on B-T4SA	**0.896**	**0.866**	**0.820**

* Approch based on low-level features
° Approch based on mid-level features
• Approch based on deep learning

Table 4. 5-Fold Cross-Validation Accuracy of different methods on Twitter Testing Dataset. *tr* stands for 'trained'; *ft* stands for 'fine-tuned'. Note that in these experiments *all* the deep models are again fine-tuned on four folds of the Twitter Testing Dataset. During cross-validation we fine-tuned all the weights of our FT models.

Figura 7: Risultati di accuratezza nella classificazione visuale del sentimento sul Twitter Testing Dataset e comparazione con lo stato dell'arte. Per i riferimenti bibliografici ed ulteriori dettagli si guardi [Vadicamo et al. 2017].

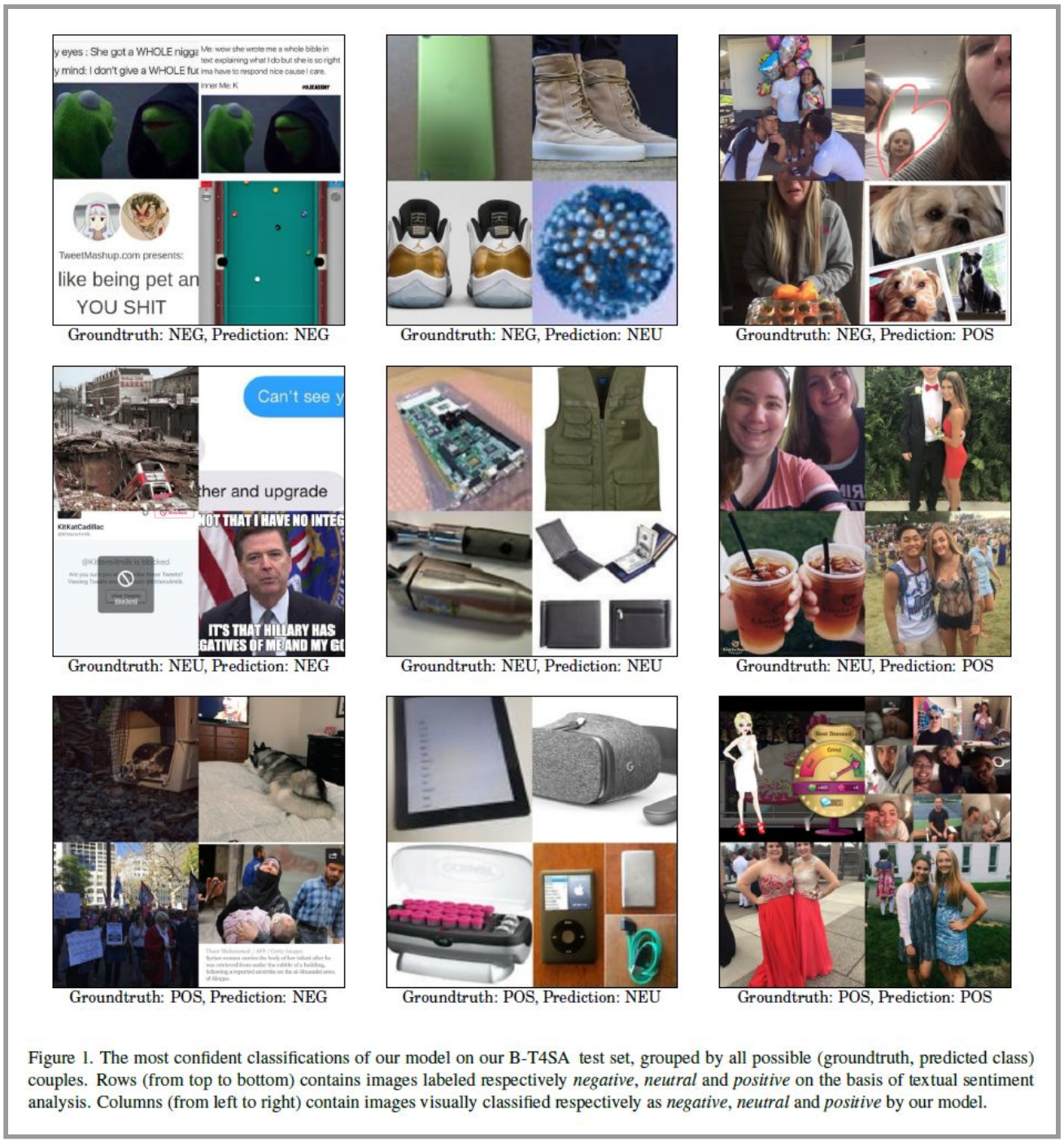


Figure 1. The most confident classifications of our model on our B-T4SA test set, grouped by all possible (groundtruth, predicted class) couples. Rows (from top to bottom) contains images labeled respectively *negative*, *neutral* and *positive* on the basis of textual sentiment analysis. Columns (from left to right) contain images visually classified respectively as *negative*, *neutral* and *positive* by our model.

Figura 8: Esempi di classificazione visuale e confronto con la predizione dedotta dal testo del tweet. Immagine estratta dalla pubblicazione [Vadicamo et al. 2017]

6 API dei Moduli di Visual Content Mining

In questa sezione, forniamo una descrizione delle API dei servizi di *Visual Content Mining* sviluppati in ambito del progetto SmartNews.

Il sommario delle chiamate REST è riportato nella Tabella 2. La Tabella 3 fornisce la descrizione dello schema degli oggetti utilizzati. Tutti i moduli accettano e producono dati in formato JSON.

Path	**Operazione**	**Descrizione**
/classify	GET	Annota una o più immagini
/clustering	GET	Raggruppa una collezione di immagini per similarità visuale
/extract_features	GET	Aggiunge una o più immagini (cioè le sue features) a un indice
/search	GET	Recupera immagini simili ad una data come query
/sentiment	GET	Analizza il sentimento visuale di una collezione di immagini

Tabella 2: Sommario delle chiamate REST.

Oggetti		
ID: integer	Identificativo di un'immagine in SmartNews. `Example: 42`	
CollectionParam: integer[]	Una lista di *ID* di immagini. `Example: [42, 156, 2525]`	
Classification: object	**id: *ID*** **classes: object[]** • class_id: string • class_label: string • score: number, $\{ x \in \mathbb{R} \mid 0 \leq x \leq 1 \}$	ID dell'immagine classificata Lista di classificazioni assegnate all'immagine Identificativo numerico della classe predetta Descrizione testuale della classe predetta Confidenza della classificazione
ClusterNode: object	**id: integer** **centroid: integer** **children: ClusterNode[]**	Identificativo locale del nodo Identificativo del nodo foglia (immagine) che è rappresentativo del gruppo I sottogruppi figli di questo nodo (che seguono di nuovo lo schema di tipo *ClusterNode*)
SentimentPrediction: object	Predizione del sentimenti di un'immagine	

	id: *ID*	ID dell'immagine analizzata
	pos: number • Example: 0.65	Confidenza della classe di sentimento 'positive'
	neg: number • Example: 0.2	Confidenza della classe di sentimento 'negative'
	neu: number • Example: 0.15	Confidenza della classe di sentimento 'neutral'
	dominant: string • Example: "pos"	Indicatore del sentimento predominante
ExtractResult: object	**id: *ID***	ID dell'immagine processata
	url: string • Example: "http://url/to/image.jpg"	URL dell'immagine processata
	status: string • Example: "ok"	Esito del processing
	message: string • Example: "success"	Messaggio di stato

Tabella 3: Definizioni dello schema degli oggetti utilizzati
Sommario delle chiamate REST.

6.1 Estrazione Feature visuali (GET /extract_features)

Aggiunge una o più immagini (cioè le sue features) a un indice. Estrae le features visuali da una o più immagini date e le aggiunge a un archivio. Le features visuali sono estratte dal livello 'PreLogits' della rete 'InceptionV3' e poi L2-normalizzate. Le immagini da processare vengono passate al modulo tramite il parametro 'id' all'interno del corpo della chiamata come una lista di stringhe, mentre il parametro 'db' indica il nome dell'archivio a cui aggiungere le immagini processate.

Parametro	**Descrizione**	**Modalità**	**Tipo**	**Schema Tipo**
id	Una lista di ID di immagini	body	object	*CollectionParam*
db	Nome dell'archivio in cui inserire le features.	query	string	

Restituisce una lista di risultati dell'operazione di indicizzazione *ExtractResult*, uno per ogni immagine. In ogni risultato viene riportato l'ID dell'immagine, la sua URL per fini di visualizzazione, lo stato del processing ('ok' o 'error') e un messaggio di descrizione del processing che può contenere ulteriori informazioni riguardo errori.

```
Esempio:
[
	{
		"id": 42,
		"url": "http://path/to/image_42.jpg",
		"status": "ok",
		"message": "success"
	},
	{
		"id": 5189,
		"url": "http://path/to/image_5189.jpg",
		"status": "ok",
		"message": "success"
	},
]
```

6.2 Analisi immagini: Annotazione testuale (GET /classify)

Annota una data collezione di immagini e ne restituisce la classificazione multi-label.

Parametro	Descrizione	Modalità	Tipo	Schema Tipo
id	Una lista di ID di immagini	body	object	*CollectionParam*
db	Nome dell'archivio da cui recuperare le immagini	query	string	

Restituisce una lista di classificazioni (*Classification*).

```
Esempio:
[
  {
    "classes": [
      {
        "class_id": "/m/09j2d",
        "score": 0.6708282232284546,
        "class_label": "clothing"
      },
      {
        "class_id": "/m/03q69",
        "score": 0.5160474181175232,
        "class_label": "hair"
      },
      {
        "class_id": "/m/0ds4x",
        "score": 0.43692442774772644,
        "class_label": "hairstyle"
      },
    ],
    "id": 3
  },
  {
    "classes": [
       {
        "class_id": "/m/01f43",
        "score": 0.5489923357963562,
        "class_label": "beauty"
      },
      {
```

```
        "class_id": "/m/0d1pc",
        "score": 0.32761526107788086,
        "class_label": "model"
      },
      {
        "class_id": "/m/05wkw",
        "score": 0.325072705745697,
        "class_label": "photography"
      },
      {
        "class_id": "/m/02zsn",
        "score": 0.3020985424518585,
        "class_label": "female"
      },
      {
        "class_id": "/m/0dzct",
        "score": 0.29410651326179504,
        "class_label": "face"
      },
      {
        "class_id": "/m/02qbl1m",
        "score": 0.29226282238960266,
        "class_label": "photo shoot"
      }
    ],
    "id": 4
  }
]
```

6.3 Analisi immagini: Sentiment (GET /sentiment)

Analizza il sentimento visuale di una collezione di immagini. Ritorna il sentimento visuale in forma di tre confidenze, rispettivamente per il sentimento positivo, negativo o neutro.

Parametro	Descrizione	Modalità	Tipo	Schema Tipo
id	Una lista di ID di immagini	body	object	*CollectionParam*
db	Nome dell'archivio da cui recuperare le immagini	query	string	

Restituisce una lista di coppie (ID, predizione del sentimento). Il formato della predizione è determinata dall'oggetto *SentimentPrediction*.

```
Esempio:
[
  {
    "neg": 0.38437220454216003,
    "neu": 0.3712303042411804,
    "dominant": "neg",
    "pos": 0.24439752101898193,
    "id": 1
  },
  {
    "neg": 0.38437220454216003,
    "neu": 0.3712303042411804,
    "dominant": "neg",
    "pos": 0.24439752101898193,
    "id": 4
  }
]
```

6.4 Analisi dall'interfaccia: Clustering (GET /clustering)

Raggruppa una collezione di immagini per similarità visuale. Clusterizza una collezione di immagini e restituisce una gerarchia (un albero binario) nella quale le foglie rappresentano le immagini date, e i nodi intermedi specificano i cluster di tali immagini.

Parametro	**Descrizione**	**Modalità**	**Tipo**	**Schema Tipo**
id	Una lista di ID di immagini	body	object	*CollectionParam*
db	Nome dell'archivio da cui recuperare le immagini	query	string	

Restituisce una gerarchia di immagini, rappresentata con un albero binario nel quale ogni nodo (gruppo) ha un ID, l'ID del suo centroide (che è sempre l'ID di una foglia), e la lista dei suoi nodi figli. L'intero albero è rappresentato dall'oggetto *ClusterNode*. Esempio:

```
Esempio:
{
	"id": 4,
	"centroid": 1,
	"children": [
	    {
		"id": 0,
		"centroid": 0
	    },
	    {
		"id": 3,
		"centroid": 1,
		"children": [
			{
				"id": 1,
				"centroid": 1
			},
			{
				"id": 2,
				"centroid": 2
			},
		]
	    }
	]
}
```

6.5 Ricerca per similarità (GET /search)

Recupera immagini simili ad una data come query. I parametri *id* e *url* sono mutualmente esclusivi. Se *id* è specificato, viene selezionata come query l'immagine presente nell'archivio avente tale id. Se *url* è specificato, viene usata come query per usare l'immagine (anche esterna a SmartNews) corrispondente alla url data.

Parametro	**Descrizione**	**Modalità**	**Tipo**
id	ID dell'immagine di query (per immagini già presenti nell'archivio)	query	integer
url	URL dell'immagine di query (se esterna a SmartNews)	query	string
limit	Quanti risultati recuperare	query	integer

Restituisce una lista di coppie (ID, score) ordinata per score decrescenti. Score alti indicano una similarità più alta con la query. Ogni coppia (ID, score) è formata da un oggetto con il seguente schema:

object

- **id:** ***ID*** — ID dell'immagine recuperata
- **score: number , { x ∈ ℝ | x ≥ 0 }** — Score di similarità

```
Esempio:
[
	{
		"id": 42,
		“score": 2314.3
	},
	{
		"id": 712,
		“score": 1248.4
	},
	{
		"id": 94,
		“score": 842.1
	}
]
```

7 Conclusioni

In questo documento sono state presentate le tecniche di visual content mining utilizzate in SmartNews per l'analisi delle immagini. In particolare, sono state presentate sia lo stato dell'arte che le soluzioni sviluppate all'interno del progetto per l'annotazione testuale, la ricerca per similarità visuale, il clustering, l'identificazione dei duplicati e la classificazione del sentimento.

I risultati dell'attività di ricerca scientifica sull'analisi delle immagini portata avanti nel progetto sono descritte in numerose pubblicazioni su atti di convegno e riviste internazionali. E' stato inoltre sviluppato il software di cinque moduli (estrazione di feature visuali, annotazione testuale, annotazione del sentiment, clustering e ricerca per similarità) che verranno integrati nella piattaforma di SmartNews.

8 Riferimenti

[Amato et al. 2016a] Amato, G., Falchi, F., & Vadicamo, L. (2016). Aggregating binary local descriptors for image retrieval. *Multimedia Tools and Applications*, 1-31.

[Amato et al. 2016b] Amato, G., Falchi, F., Gennaro, C., & Vadicamo, L. (2016). Deep permutations: deep convolutional neural networks and permutation-based indexing. In *International Conference on Similarity Search and Applications* (pp. 93-106). Springer International Publishing..

[Amato et al. 2016c] Amato G., Falchi, Gennaro C., F Rabitti. YFCC100M-HNfc6: a large-scale deep features benchmark for similarity search. In International Conference on Similarity Search and Applications, 196-209

[Amato et al. 2016d] Amato, G., Debole, F., Falchi, F., Gennaro, C., & Rabitti, F. (2016, September). Large scale indexing and searching deep convolutional neural network features. In *International Conference on Big Data Analytics and Knowledge Discovery* (pp. 213-224). Springer, Cham.

[Amato et al. 2016e] Amato, G., Falchi, F., Gennaro, C., & Rabitti, F. (2016, October). YFCC100M HybridNet fc6 Deep Features for Content-Based Image Retrieval. In *Proceedings of the 2016 ACM Workshop on Multimedia COMMONS* (pp. 11-18). ACM.

[Amato et al. 2017] Amato, G., Falchi, F., Gennaro, C., & Rabitti, F. (2017, June). Searching and annotating 100M Images with YFCC100M-HNfc6 and MI-File. In *Proceedings of the 15th International Workshop on Content-Based Multimedia Indexing* (p. 26). ACM.

[Amato et al. 2014] Amato, G., Gennaro, C., & Savino, P. (2014). MI-File: using inverted files for scalable approximate similarity search. *Multimedia tools and applications*, *71*(3), 1333-1362.

[Babenko et al. 2014] Babenko, A., Slesarev, A., Chigorin, A., & Lempitsky, V. (2014, September). Neural codes for image retrieval. In *European conference on computer vision*(pp. 584-599). Springer, Cham.

[Bay et al. 2006] Bay, H., Tuytelaars, T., & Van Gool, L. (2006). Surf: Speeded up robust features. *Computer vision–ECCV 2006*, 404-417.

[Bawa et al. 2005] Bawa, M., Condie, T., & Ganesan, P. (2005, May). LSH forest: self-tuning indexes for similarity search. In *Proceedings of the 14th international conference on World Wide Web* (pp. 651-660). ACM.

[Borth et al. 2103] Borth, D., Ji, R., Chen, T., Breuel, T., & Chang, S. F. (2013, October). Large-scale visual sentiment ontology and detectors using adjective noun pairs. In *Proceedings of the 21st ACM international conference on Multimedia* (pp. 223-232). ACM.

[Campos et al. 2017] Campos, V., Jou, B., & Giro-i-Nieto, X. (2017). From pixels to sentiment: Fine-tuning cnns for visual sentiment prediction. *Image and Vision Computing*.

[Cao et al. 2016] Cao, D., Ji, R., Lin, D., & Li, S. (2016). A cross-media public sentiment analysis system for microblog. *Multimedia Systems*, *22*(4), 479-486.

[Carrara et al. 2017] Carrara F., Esuli A., Fagni T., Falchi F., Fernández A.M., Picture it in your mind: Generating high level visual representations from textual descriptions. In Information Retrieval Journal, 1-22

[Carrara et al. 2017b] Carrara, F., Falchi, F., Caldelli, R., Amato, G., Fumarola, R., & Becarelli, R. (2017, June). Detecting adversarial example attacks to deep neural networks. In *Proceedings of the 15th International Workshop on Content-Based Multimedia Indexing* (p. 38). ACM.

[Chavez et al. 2008] Chavez, E., Figueroa, K., & Navarro, G. (2008). Effective proximity retrieval by ordering permutations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *30*(9), 1647-1658.

[Chen et al. 2014] Chen, T., Borth, D., Darrell, T., & Chang, S. F. (2014). Deepsentibank: Visual sentiment concept classification with deep convolutional neural networks. *arXiv preprint arXiv:1410.8586*.

[Cimino & Dell'Orletta 2016] Cimino, A., & Dell'Orletta, F. (2016). Tandem LSTM-SVM Approach for Sentiment Analysis. In *CLiC-it/EVALITA*.

[Connor et al. 2017] Connor, R., Vadicamo, L., & Rabitti, F. (2017, October). High-Dimensional Simplexes for Supermetric Search. In *International Conference on Similarity Search and Applications* (pp. 96-109). Springer, Cham.

[Conner et al. 2018] Connor, R., Vadicamo, L., Cardillo, F. A., & Rabitti, F. (2018). Supermetric Search. Information Systems.

[Datta et al. 2005] Datta, R., Li, J., & Wang, J. Z. (2005, November). Content-based image retrieval: approaches and trends of the new age. In *Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval* (pp. 253-262). ACM

[Deng & Manjunath 2001] Deng, Y., & Manjunath, B. S. (2001). Unsupervised segmentation of color-texture regions in images and video. *IEEE transactions on pattern analysis and machine intelligence*, *23*(8), 800-810.

[Friedman et al. 1977] Friedman, J. H., Bentley, J. L., & Finkel, R. A. (1977). An algorithm for finding best matches in logarithmic expected time. *ACM Transactions on Mathematical Software (TOMS)*, *3*(3), 209-226.

[Galvez-Lopez & Tardos 2011] Galvez-Lopez, D., & Tardos, J. D. (2011, September). Real-time loop detection with bags of binary words. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on* (pp. 51-58). IEEE.

[Grana et al. 2013] Grana, C., Borghesani, D., Manfredi, M., & Cucchiara, R. (2013, March). A fast approach for integrating ORB descriptors in the bag of words model. In *Proc. SPIE* (Vol. 8667, pp. 866709-866709).

[Guo et al. 2016] Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S., & Lew, M. S. (2016). Deep learning for visual understanding: A review. *Neurocomputing*, *187*, 27-48.

[Hare et al. 2006] Hare, J. S., Lewis, P. H., Enser, P. G., & Sandom, C. J. (2006). Mind the Gap: Another look at the problem of the semantic gap in image retrieval.

[Indyk & Motwani 1998] Indyk, P., & Motwani, R. (1998, May). Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing* (pp. 604-613). ACM.

[Islam & Zhang 2016] Islam, J., & Zhang, Y. (2016, October). Visual Sentiment Analysis for Social Images Using Transfer Learning Approach. In *Big Data and Cloud Computing (BDCloud), Social Computing and Networking (SocialCom), Sustainable Computing and Communications (SustainCom)(BDCloud-SocialCom-SustainCom), 2016 IEEE International Conferences on* (pp. 124-130). IEEE.

[Krizhevsky et al. 2012] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).

[Kodituwakku & Selvarajah 2004] Kodituwakku, S. R., & Selvarajah, S. (2004). Comparison of color features for image retrieval. *Indian Journal of Computer Science and Engineering*, *1*(3), 207-211.

[Jégou et al. 2008] Jegou, H., Douze, M., & Schmid, C. (2008, October). Hamming embedding and weak geometric consistency for large scale image search. In *European conference on computer vision* (pp. 304-317). Springer, Berlin, Heidelberg.

[Jégou et al. 2010a] Jégou, H., Douze, M., Schmid, C., & Pérez, P. (2010, June). Aggregating local descriptors into a compact image representation. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on* (pp. 3304-3311). IEEE.

[Jégou et al. 2010b] Jégou, H., Douze, M., Schmid, C., & Pérez, P. (2010, June). Aggregating local descriptors into a compact image representation. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on* (pp. 3304-3311). IEEE.

[Jou et al. 2015] Jou, B., Chen, T., Pappas, N., Redi, M., Topkara, M., & Chang, S. F. (2015, October). Visual affect around the world: A large-scale multilingual visual sentiment ontology. In *Proceedings of the 23rd ACM international conference on Multimedia* (pp. 159-168). ACM.

[LeCun et al. 2015] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436-444.

[Lee et al. 2015] Lee, S., Choi, S., & Yang, H. S. (2015). Bag-of-binary-features for fast image representation. *Electronics Letters*, *51*(7), 555-557.

[Li et al. 2011] Li, S., Wang, Z., Zhou, G., & Lee, S. Y. M. (2011, June). Semi-supervised learning for imbalanced sentiment classification. In *IJCAI proceedings-international joint conference on artificial intelligence*(Vol. 22, No. 3, p. 1826).

[Li et al. 2016] Li, X., Uricchio, T., Ballan, L., Bertini, M., Snoek, C. G., & Bimbo, A. D. (2016). Socializing the semantic gap: A comparative survey on image tag assignment, refinement, and retrieval. *ACM Computing Surveys (CSUR)*, *49*(1), 14.

[Li et al. 2017] Li, Z., Fan, Y., Liu, W., & Wang, F. (2017). Image sentiment prediction based on textual descriptions with adjective noun pairs. *Multimedia Tools and Applications*, 1-18.

[Liu et al. 2007] Liu, Y., Zhang, D., Lu, G., & Ma, W. Y. (2007). A survey of content-based image retrieval with high-level semantics. *Pattern recognition*, *40*(1), 262-282.

[Lowe 2004] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, *60*(2), 91-110.

[Lv et al. 2007] Lv, Q., Josephson, W., Wang, Z., Charikar, M., & Li, K. (2007, September). Multi-probe LSH: efficient indexing for high-dimensional similarity search. In *Proceedings of the 33rd international conference on Very large data bases* (pp. 950-961). VLDB Endowment.

[Machajdik & Hanbury 2010] Machajdik, J., & Hanbury, A. (2010, October). Affective image classification using features inspired by psychology and art theory. In *Proceedings of the 18th ACM international conference on Multimedia*(pp. 83-92). ACM.

Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval* (Vol. 1, No. 1, p. 496). Cambridge: Cambridge university press.

[Manjunath et al. 2002] Manjunath, B. S., Salembier, P., & Sikora, T. (Eds.). (2002). *Introduction to MPEG-7: multimedia content description interface* (Vol. 1). John Wiley & Sons.

[Mehrotra & Gary 1995] Mehrotra, R., & Gary, J. E. (1995). Similar-shape retrieval in shape data management. *Computer*, *28*(9), 57-62.

[Novak & Zezula 2016] Novak, D., & Zezula, P. (2016). PPP-codes for large-scale similarity searching. In *Transactions on Large-Scale Data-and Knowledge-Centered Systems XXIV*(pp. 61-87). Springer Berlin Heidelberg.

[Novak et al. 2011] Novak, D., Batko, M., & Zezula, P. (2011). Metric index: An efficient and scalable solution for precise and approximate similarity search. *Information Systems*, *36*(4), 721-733.

[Park et al. 2002] Park, M., Jin, J. S., & Wilson, L. S. (2002). Fast content-based image retrieval using quasi-gabor filter and reduction of image feature dimension. In *Image Analysis and Interpretation, 2002. Proceedings. Fifth IEEE Southwest Symposium on* (pp. 178-182). IEEE.

[Perronnin & Dance 2007] Perronnin, F., & Dance, C. (2007, June). Fisher kernels on visual vocabularies for image categorization. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*(pp. 1-8). IEEE.

[Perronnin et al. 2010a] Perronnin, F., Liu, Y., Sánchez, J., & Poirier, H. (2010, June). Large-scale image retrieval with compressed fisher vectors. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on* (pp. 3384-3391). IEEE.

[Perronnin et al. 2010b] Perronnin, F., Sánchez, J., & Mensink, T. (2010). Improving the fisher kernel for large-scale image classification. *Computer Vision–ECCV 2010*, 143-156.

[Philbin et al. 2007] Philbin, J., Chum, O., Isard, M., Sivic, J., & Zisserman, A. (2007, June). Object retrieval with large vocabularies and fast spatial matching. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on* (pp. 1-8). IEEE.

[Rao et al. 2016] Rao, T., Xu, M., & Xu, D. (2016). Learning Multi-level Deep Representations for Image Emotion Classification. *arXiv preprint arXiv:1611.07145*.

[Razavian et al. 2014] Sharif Razavian, A., Azizpour, H., Sullivan, J., & Carlsson, S. (2014). CNN features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops* (pp. 806-813).

[Rublee et al. 2016] Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011, November). ORB: An efficient alternative to SIFT or SURF. In *Computer Vision (ICCV), 2011 IEEE international conference on* (pp. 2564-2571). IEEE.

[Siersdorfer et al. 2010a] Siersdorfer, S., Minack, E., Deng, F., & Hare, J. (2010, October). Analyzing and predicting sentiment of images on the social web. In *Proceedings of the 18th ACM international conference on Multimedia* (pp. 715-718). ACM.

[Siersdorfer et al. 2010b] Siersdorfer, S., Minack, E., Deng, F., & Hare, J. (2010, October). Analyzing and predicting sentiment of images on the social web. In *Proceedings of the 18th ACM international conference on Multimedia*(pp. 715-718). ACM.

[Simonyan & Zisserman 2014] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[Sivic & Zisserman 2003] Sivic, J., & Zisserman, A. (2003, October). Video Google: A text retrieval approach to object matching in videos. In *null* (p. 1470). IEEE.

[Smeulders et al. 2000] Smeulders, A. W., Worring, M., Santini, S., Gupta, A., & Jain, R. (2000). Content-based image retrieval at the end of the early years. *IEEE Transactions on pattern analysis and machine intelligence*, *22*(12), 1349-1380.

[Uchida et al. 2016] Uchida, Y., Sakazawa, S., & Satoh, S. I. (2016). Image retrieval with fisher vectors of binary features. *ITE Transactions on Media Technology and Applications*, *4*(4), 326-336.

[Vadicamo et al. 2017] Vadicamo, L., Carrara, F., Cimino, A., Cresci, S., Dell'Orletta, F., Falchi, F., & Tesconi, M. (2017). *Cross-Media Learning for Image Sentiment Analysis in the Wild*. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 308-317)

[Van Opdenbosch et al. 2014] Van Opdenbosch, D., Schroth, G., Huitl, R., Hilsenbeck, S., Garcea, A., & Steinbach, E. (2014, October). Camera-based indoor positioning using scalable streaming of compressed binary image signatures. In *Image Processing (ICIP), 2014 IEEE International Conference on* (pp. 2804-2808). IEEE.

[Wang et al. 2015] Wang, Y., Wang, S., Tang, J., Liu, H., & Li, B. (2015, July). Unsupervised Sentiment Analysis for Social Media Images. In *IJCAI* (pp. 2378-2379).

[Ward 1963] Ward Jr, J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American statistical association*, *58*(301), 236-244.

[You et al. 2015] You, Q., Luo, J., Jin, H., & Yang, J. (2015, January). Robust Image Sentiment Analysis Using Progressively Trained and Domain Transferred Deep Networks. In *AAAI* (pp. 381-388).

[You et al. 2016] You, Q., Luo, J., Jin, H., & Yang, J. (2016, February). Cross-modality consistent regression for joint visual-textual sentiment analysis of social multimedia. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining* (pp. 13-22). ACM.

[Yuan et al 2013] Yuan, J., Mcdonough, S., You, Q., & Luo, J. (2013, August). Sentribute: image sentiment analysis from a mid-level perspective. In *Proceedings of the Second International Workshop on Issues of Sentiment Discovery and Opinion Mining* (p. 10). ACM.

[Zezula et al. 2006] Zezula, P., Amato, G., Dohnal, V., & Batko, M. (2006). *Similarity search: the metric space approach*(Vol. 32). Springer Science & Business Media.

[Zhang et al. 2013] Zhang, Y., Zhu, C., Bres, S., & Chen, L. (2013, March). Encoding Local Binary Descriptors by Bag-of-Features with Hamming Distance for Visual Object Categorization. In *ECIR* (pp. 630-641).

[Zhou & Huang 2003] Zhou, X. S., & Huang, T. S. (2003). Relevance feedback in image retrieval: A comprehensive review. *Multimedia systems*, *8*(6), 536-544.

[Zhou et al. 2014] Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., & Oliva, A. (2014). Learning deep features for scene recognition using places database. In *Advances in neural information processing systems* (pp. 487-495).