دانشگاه سمنان

KEY.NET: KEYPOINT DETECTION BY HANDCRAFTED AND LEARNED CNN FILTERS

DR. FADAEIESLAM

AMIR SHOKRI

ALIREZA GHOLAMNIA

FARSHAD ASGHARZADE

دانشگاه سمنان

هر کس به زبانی تو را حمد گوید

TABLE OF CONTENTS

TABLE OF CONTENTS

ABSTRACT

- keypoint detection
- handcrafted and learned CNN filters
- multi-scale architecture
- Handcrafted filters provide anchor structures for learned filters, which localize, score and rank repeatable features.
- loss function
- Data :
  - ImageNet and evaluated on HPatches benchmark.
- outperforms state-of-the-art detectors in terms of repeatability, matching performance and complexity.

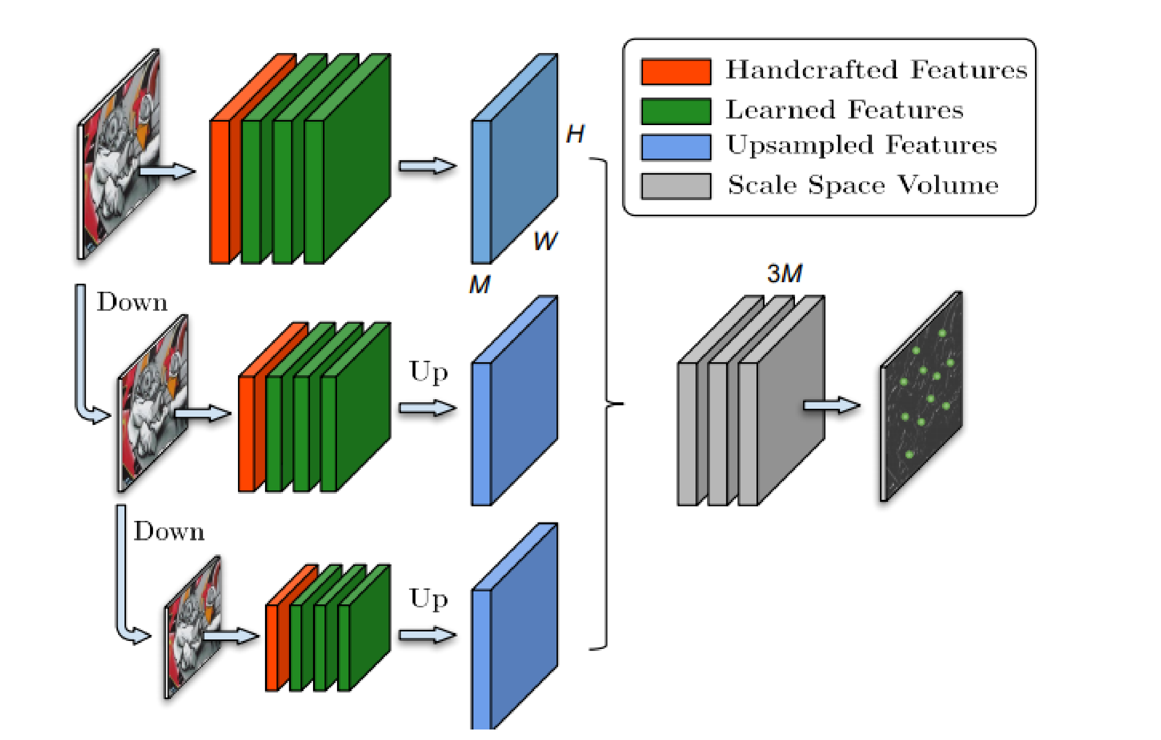
INTRODUCTION

- self-guided navigation or 3D reconstruction
- advantage of learning methods over handcrafted ones has not been clearly demonstrated in keypoint detection.
- Convolutional Neural Networks (CNNs)
- local feature detectors were based on engineered filters. For instance, approaches such as Difference of Gaussians, Harris-Laplace or Hessian-Affine use combinations of image derivatives to compute feature maps, which is remarkably similar to the operations in trained CNN’s layers.

INTRODUCTION

- In summary, our contributions are the following:
  - a) a keypoint detector that combines handcrafted and learned CNN features
  - b) a novel multi-scale loss and operator for detecting and ranking stable keypoints across scales,
  - c) a multi-scale feature detection with shallow architecture.

INTRODUCTION



تصویر ۱ : معماری پیشنهادی *Key.Net* ترکیبی از فیلترهای ساخته شده و آموخته شده برای استخراج ویژگی ها در مقیاس های مختلف است. نقشه های ویژگی نمونه برداری و منشور می شوند. فیلتر آخرین آموخته شده برای به دست آوردن *final response map* ، میزان فضای مقیاس را ترکیب می کند.

RELATED WORK

We present related works in two main categories:

- Handcrafted
- learned based

HANDCRAFTED DETECTORS

Traditional feature detectors localize geometric structures through engineered algorithms, which are often referred to as handcrafted. Harris and Hessian detectors used first and second order image derivatives to find corners or blobs in images.

Multi-scale improvements were proposed in KAZE and its extension, A-KAZE, where Hessian detector was applied to a non-linear diffusion scale space in contrast to widely used Gaussian pyramid.

Although corner detectors proved to be robust and efficient, other methods seek lternative structures within images. SIFT looked for blobs over multiple scale levels, and MSER segmented and selected stable regions as keypoints.

LEARNED DETECTORS

Latest advances in CNNs also made an impact on feature detection. TILDE trained multiple piece-wise linear regression models to identify interest points that are robust under severe weather and illumination changes. introduced a new formulation to train a CNN based on feature covariant constraints. Previous detector was extended in by adding predefined detector anchors, showing improved stability in training.

Latest advances in CNNs also made an impact on feature detection. TILDE trained multiple piece-wise linear regression models to identify interest points that are robust under severe weather and illumination changes.

introduced a new formulation to train a CNN based on feature covariant constraints. Previous detector was extended in by adding predefined detector anchors, showing improved stability in training.

LEARNED DETECTORS

Quadruple image patches and a ranking scheme of point responses as cost function were used in to train a neural network. In, authors proposed a pipeline to automatically sample positive and negative pairs of patches from a region proposal network to optimize jointly point detections and their representations. Recently, LF-Net estimated position, scale and orientation of features by optimizing jointly the detector and descriptor.

KEY.NET ARCHITECTURE

Key.Net architecture combines successful ideas from handcrafted and learned methods namely gradient-based feature extraction, learned combinations of low-level features and multi-scale pyramid representation.

HANDCRAFTED AND LEARNED FILTERS

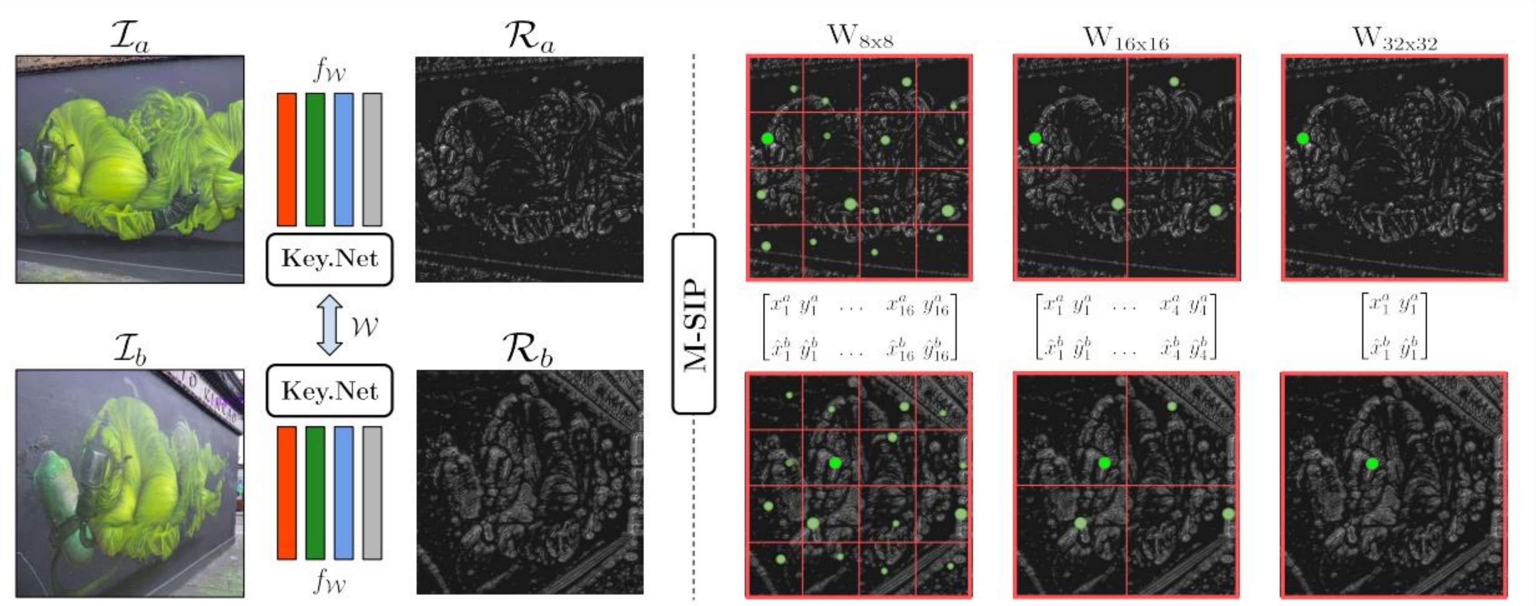
$$I_{i_1,\dots,i_n} = I_0 * \partial_{i_1},\dots,{i_n} g\sigma(\underset{x}{\rightarrow})$$

جایی که g_σنشانگر Gaussian با عرض σ است که در $\underset{0}{\rightarrow} = \underset{x}{\rightarrow}$ قرار دارد و جهت را نشان می دهد. مشتقات مرتبه بالاتر یعنی n>2 به نویز حساس هستند و به هسته های بزرگ احتیاج دارند ، بنابراین مشتقات و ترکیبات آنها را فقط تا مرتبه دوم شامل می کنیم:

- First order : از تصویر I شیب درجه یک I_x و I_y را بدست می آوریم. علاوه بر این ، ما $I_y * I_x$، $I_x{}^2$ و $I_y{}^2$ را در ماتریس لحظه دوم ردیاب هریس محاسبه می کنیم.
- Second order : از تصویر I ، مشتقات مرتبه ۲ ، I_{xx}، I_{yy} و I_{xy} نیز مانند ماتریس هسیان استفاده شده در آشکارسازهای هسیان و DoG هستند. از آنجا که آشکارساز هسی از تعیین کننده ماتریس هسی استفاده می کند ، $I_{yy} * I_{xx}$ و $I^2{}_{xy}$ را اضافه می کنیم.
- Learned : یک لایه کانولوشن با فیلترهای M ، یک لایه نرمال سازی دسته ای و یک تابع فعال سازی ReLUیک بلوک یاد گرفته شده را تشکیل می دهد.

فیلترهای رمزگذاری شده سخت، تعداد پارامترهای قابل یادگیری برای آموزش معماری را کاهش می دهند و باعث بهبود ثبات و همگرایی در حین تولید مجدد می شوند.

HANDCRAFTED AND LEARNED FILTERS



تصویر ۲ : روند آموزش سیامی. تصویر I_a و I_b برای تولید نقشه های پاسخ خود ، R_a و R_b از طریق *Key.Net* می روند. *M-SIP* مختصات نقطه علاقه را برای هر یک از پنجره ها در مناطق چند مقیاس پیشنهاد می کند. تابع از دست دادن نهایی به عنوان یک رگرسیون از شاخص های مختصات از I_a و حداکثر مختصات محلی از I_b محاسبه می شود. بهتر تجسم رنگ است.

MULTI-SCALE PYRAMID

e robust to small scale changes without the need for computing several forward passes.

All the feature maps resulting from the handcrafted filters are concatenated to feed the stack of learned filters in each of the scale levels.

All three streams share the weights, such that the same type of anchors result from different levels and form the set of candidates for final keypoints. Feature maps from all scale levels are then upsampled, concatenated and fed to the last convolutional filter to obtain the final response map.

LOSS FUNCTIONS

In supervised training, the loss function relies on the ground truth.

In the case of keypoints, ground truth is not well defined as keypoint locations are useful as long as they can be accurately detected regardless of geometric or photometric image transformation.

chors to guide their training. Although anchors make the training more stable and lead to better results, they prevent the network from proposing new keypoints in case there is no anchor in the proximity.

In contrast, the handcrafted filters in Key.Net provide a weak constraint with the benefit of the anchor-based methods while allowing the detector to propose new stable keypoints. In our approach, only the geometric transformation between images is required to guide the loss.

INDEX PROPOSAL LAYER

$$m_i(u, v) = \frac{e^{w_i(u,v)}}{\Sigma_{j,k}^{N} e^{w_I}(j, k)}$$

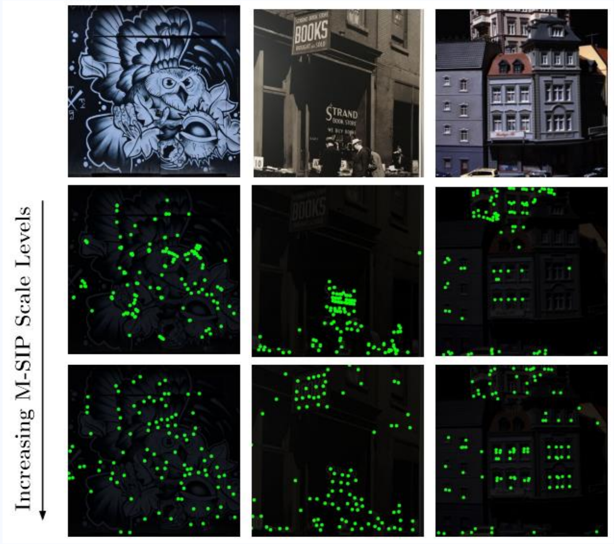
$$[x_i, y_i]^T = [\,\bar{u_i}\,, \bar{v_i}]^T = \Sigma_{u,v}^{T}\,[W \odot m_i, W^T \odot m_i]^T + c_w$$

INDEX PROPOSAL LAYER

$$\mathcal{L}_{IP}\left(I_a, I_b, H_{a,b}, N\right) = \Sigma_i \alpha_i \,||\, [\, x_i, y_i \,]_a^T - H_{b,a}[\widehat{x_i}, \widehat{y_i}]_b^T \,||^2$$

$$\alpha_i = R_a(x_i, y_i)_a + R_b(\widehat{x_i}, \widehat{y_i})_b$$

MULTI-SCALE PROPOSAL LAYER



تصویر ۳ : نکات کلیدی پس از افزودن پنجره های بزرگتر به اپراتور M-SIP بدست می آیند. نقاطی که پایدارتر هستند همچنان که اپراتور M-SIP اندازه پنجره خود را افزایش می دهد ، باقی می مانند. نقشه های مشخصه در ردیف میانی حاوی نقاطی در اطراف لبه ها یا مناطق غیرمتمایز است ، در حالی که ردیف پایین تشخیص هایی را نشان می دهد که در زیر تبدیلات هندسی قوی تر هستند.

MULTI-SCALE INDEX PROPOSAL LAYER

$$\mathcal{L}_{MSIP}\left(I_a, I_b, H_{a,b}\right) = \Sigma_s \lambda_s \mathcal{L}_{IP}(I_a, I_b, H_{a,b}, N_s)$$

EXPERIMENTAL SETTINGS

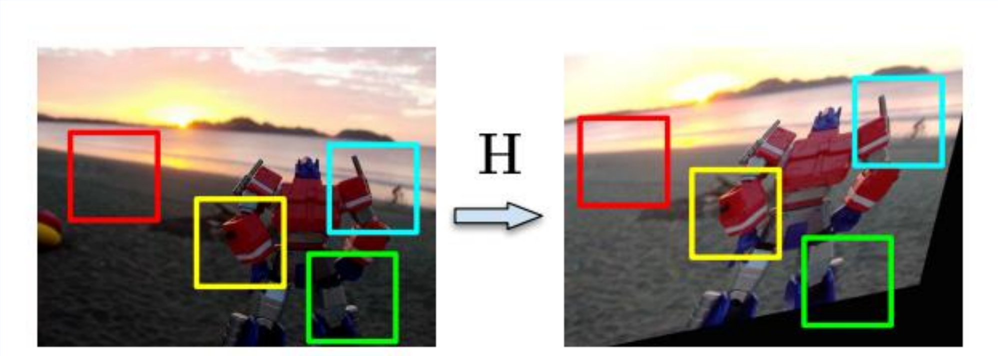
In this section, we present implementation details, metrics and the dataset used for evaluating the method.

TRAINING DATA

ImageNet ILSVRC 2012 dataset

There are 12,000 image pairs of size 192 × 192. We use 9,000 of them as the training data and 3,000 as validation set.

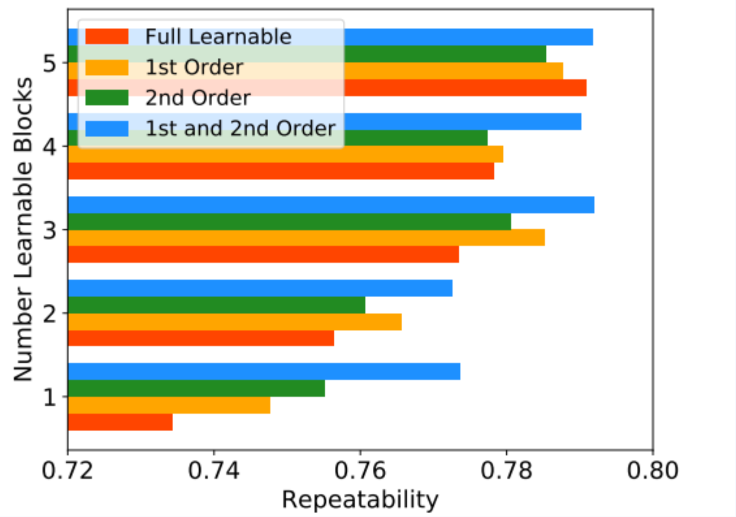
PRELIMINARY ANALYSIS



تصویر ۴ : ما تحولات هندسی و فوتومتریک تصادفی را به تصاویر اعمال می کنیم و به عنوان مجموعه آموزش ، جفت مناطق متناظر را استخراج می کنیم. منطقه قرمز با بررسی پاسخ فیلترهای دست ساز کنار گذاشته می شود.

IMPLEMENTATION NOTES

M-SIP Region Sizes					
W_{8x8}	W_{16x16}	W_{24x24}	W_{32x32}	W_{40x40}	Repeatability
✓	-	-	-	-	70.5
✓	✓	-	-	-	74.6
✓	✓	✓	-	-	76.8
✓	✓	✓	✓	-	77.6
-	-	-	-	✓	65.7
-	-	-	✓	✓	71.4
-	-	✓	✓	✓	73.2
-	✓	✓	✓	✓	74.9
✓	✓	✓	✓	✓	**79.1**



تصویر ۵ : چپ: مقایسه نتایج تکرارپذیری برای چندین سطح در عملگر M-SIP. ما ترکیبات مختلفی از ضررهای زمینه را به عنوان ضرر نهایی ، از مناطق کوچکتر به بزرگتر نشان می دهیم. بهترین نتیجه هنگام استفاده از پنج اندازه پنجره از ۸ × ۸ تا ۴۰ ۴۰ ۴۰ ۴۰ است. درست: نتایج تکرار برای ترکیبات مختلف فیلترهای

handcraftedو تعدادی لایه قابل یادگیری (هر کدام ۸ فیلتر M =). تعداد بیشتری از لایه ها منجر به نتایج بهتر می شوند. تمام نمرات تکرارپذیری براساس اعتبارسنجی مصنوعی تنظیم شده از ImageNet محاسبه می شوند.

RESULTS

In this section, we present the experiments and discuss the results. We first show results on validation data for several variants of the proposed architecture. Next, Key.Net repeatability scores in single-scale and multi-scale are presented along with the state-of-the art detectors on HPatches. Moreover, we evaluate the matching performance, the number of learnable parameters and inference time of our proposed detector and compare to other techniques.

PRELIMINARY ANALYSIS

	Num. Pyramid Levels					
	1	2	3	4	5	6
Rep.	72.5	74.6	79.1	79.4	**79.5**	78.6

(a) Number of input scale levels in Key.Net.

	Spatial Softmax Base					
	1.2	1.4	2.0	e	5.0	7.5
Rep.	77.5	78.4	77.9	**79.1**	74.6	73.2

(b) Spatial softmax base used in equation 2.

جدول ۱ : نتایج تکرار برای گزینه های مختلف طراحی در مجموعه اعتبار سنجی تنظیم شده از ImageNet.

PRELIMINARY ANALYSIS

M-SIP Levels

Filter Combinations

Multiple Pyramid Levels

Spatial Softmax Base

KEYPOINT DETECTION

	Viewpoint					Illumination				
	Repeatability		$\bar{\epsilon}_{IoU}$		S_{range}	Repeatability		$\bar{\epsilon}_{IoU}$		S_{range}
	SL	L	SL	L	SL	SL	L	SL	L	SL
SIFT-SI [5]	43.1	57.6	0.18	0.12	78.6	47.8	60.4	0.18	0.12	84.5
SURF-SI [20]	46.7	60.3	0.18	0.18	24.8	53.0	64.0	0.15	0.11	27.4
FAST-TI [24]	30.4	63.1	0.21	0.10	-	63.6	63.6	**0.09**	0.09	-
MSER-SI [23]	56.4	62.8	**0.12**	**0.08**	**503.7**	46.5	54.5	0.12	0.10	**524.8**
Harris-Laplace-SI [34]	45.1	62.0	0.20	0.13	95.9	52.7	62.0	0.17	**0.08**	90.4
KAZE-SI [21]	53.3	65.7	0.20	0.11	12.5	56.9	65.7	0.12	0.10	12.7
AKAZE-SI [22]	54.0	65.6	0.19	0.10	13.5	64.9	69.1	0.11	0.09	13.6
TILDE-TI [14]	31.0	65.1	0.20	0.15	-	70.4	70.4	0.11	0.11	-
LIFT-SI [7]	43.4	59.4	0.20	0.13	13.3	51.6	65.4	0.18	0.12	13.8
DNet-SI [9]	49.4	62.2	0.21	0.14	11.4	59.1	65.1	0.14	0.13	17.1
TCDET-SI [10]	49.6	61.6	0.23	0.16	6.7	66.9	71.0	0.16	0.15	11.4
SuperPoint-TI [13]	33.3	67.1	0.20	0.17	-	69.9	69.9	0.10	0.10	-
LF-Net-SI [11]	32.3	62.2	0.23	0.12	2.00	68.6	69.1	0.10	0.10	2.0
Tiny-Key.Net-SI	57.8	70.3	0.20	0.12	7.6	56.1	62.8	0.14	0.11	7.6
Key.Net-TI	34.2	71.5	0.20	0.11	-	**72.0**	**72.0**	0.10	0.10	-
Key.Net-SI	**60.5**	**73.2**	0.19	0.14	7.6	61.3	66.2	0.12	0.10	7.6

جدول ۲ : نتایج تکرارپذیری (٪) برای ترجمه (TI) و مقیاس (SI) آشکارسازهای ثابت در HPatches. ما همچنین خطای متوسط همپوشانی $\bar{IoU}$ و نسبت حداکثر به حداقل مقیاس استخراج شده SRange را گزارش می دهیم. در SL ، مقیاس ها و مکان ها برای محاسبه خطای همپوشانی استفاده می شود ، در عین حال ، در L ، فقط مکان ها استفاده می شوند و مقیاس ها به درستی تخمین زده می شوند. Key.Net و TinyKey.Net برای L و SL بهترین الگوریتم های دیدگاه هستند. در توالی های روشنایی ، Key.Net-TI بی تغییر ترجمه بهترین دقت را به دست می آورد. در میان آشکارسازهای ثابت SI ثابت ، TCDET بهترین در L و LF-Net در SL است.

KEYPOINT MATCHING

	Matching Score	
	View	Illum
MSER [23] + HardNet [38]	11.7	18.8
SIFT [5] + HardNet [38]	23.2	24.8
HarrisLaplace [34] + HardNet [38]	30.0	31.7
AKAZE [22] + HardNet [38]	36.4	41.4
TILDE [14] + HardNet [38]	32.3	39.3
LIFT [7] + HardNet [38]	30.3	32.8
DNet [9] + HardNet [38]	33.5	34.7
TCDET [10] + HardNet [38]	27.6	36.3
SuperPoint [13] + HardNet [38]	37.4	<u>43.0</u>
LF-Net [11] + HardNet [38]	26.9	**43.8**
LIFT [7]	21.8	26.5
SuperPoint [13]	<u>38.0</u>	41.5
LF-Net [11]	23.0	29.1
Tiny-Key.Net + HardNet [38]	37.9	37.3
Key.Net + HardNet [38]	**38.4**	39.7

جدول ۳ : نمره تطبیق (٪) بهترین آشکارسازها با HardNet و پیشرفته ترین ردیاب ها / توصیف کننده ها. نتایج در توالی HPatches ، هم از نظر دید و هم از نظر میزان روشنایی. معماری Key.Net بهترین نمره تطبیق را برای viewpoint کسب می کند ، در حالی که LF-Net + HardNet برای توالی های روشنایی است.

EFFICIENCY

Number of Learnable Parameters				
TCDET	SuperPoint	LF-Net	Tiny-Key.Net	Key.Net
548k	940k	39k	**280**	5.9k

جدول ۴ : مقایسه تعداد پارامترهای قابل یادگیری برای معماری های پیشرفته. *Tiny-Key.Net* فقط یک بلوک قابل یادگیری با یک فیلتر دارد.

EFFICIENCY

We also compare the number of learnable parameters, indicating then the complexity of the predictor, which leads to an increasing risk of overfitting and need for a large amount of training data. Table 4 shows the approximate number of parameters for different architectures. Learnable parameters that are not used during inference in the detector part are not counted for SuperPoint and LF-Net detectors. The highest complexity is from SuperPoint with 940k learnable parameters. Key.Net has nearly 160 times fewer parameters and Tiny-Key.Net has 3,100 times fewer parameters than SuperPoint with better repeatability for viewpoint scenes. The inference time of an image of 600 × 600 is 5.7ms (175 FPS) and 31ms (32.25 FPS) for Tiny-Key.Net and Key.Net, respectively.

CONCLUSIONS

We have introduced a novel approach to detect local features that combines handcrafted and learned CNN filters. We have proposed a multi-scale index proposal layer that finds keypoints across a range of scales, with a loss function that optimizes the robustness and discriminating properties of the detections. We demonstrated how to compute and combine differentiable keypoint detection loss for multiscale representation. Evaluation results on large benchmark show that combining handcrafted and learned features as well as multi-scale analysis at different stages of the network improves the repeatability scores compared to other state-of-the-art keypoint detection methods.

CONCLUSIONS

We further show that excessively increasing network's complexity does not lead to improved results. In contrast, using handcrafted filters allows to significantly reduce the complexity of the architecture leading to a detector with 280 learnable parameters and inference of 175 frames per second. Proposed detectors lead to state of the-art matching performance when used with a descriptor on viewpoint.

REFERENCES

[1] Karel Lenc and Andrea Vedaldi. Large scale evaluation of local image feature detectors on homography datasets. BMVC, 2018.

[2] Vassileios Balntas, Karel Lenc, Andrea Vedaldi, and Krystian Mikolajczyk. Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. CVPR, 2017.

[3] Xufeng Han, Thomas Leung, Yangqing Jia, Rahul Sukthankar, and Alexander C. Berg. Matchnet: Unifying feature and metric learning for patch-based matching. CVPR, 2015.

[4] Sergey Zagoruyko and Nikos Komodakis. Learning to compare image patches via convolutional neural networks. CVPR, 2015.

[5] David G. Lowe. Distinctive image features from scaleinvariant keypoints. IJCV, 2004.

[6] Krystian Mikolajczyk and Cordelia Schmid. Scale & affine invariant interest point detectors. ICCV, 2004.

[7] Kwang Moo Yi, Eduard Trulls, Vincent Lepetit, and Pascal Fua. Lift: Learned invariant feature transform. ECCV, 2016.

[8] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Toward geometric deep slam. arXiv preprint arXiv:1707.07410, 2017.

[9] Karel Lenc and Andrea Vedaldi. Learning covariant feature detectors. ECCV, 2016.

[10] Xu Zhang, Felix X. Yu, Svebor Karaman, and Shih-Fu Chang. Learning discriminative and transformation covariant local feature detectors. CVPR, 2017.

[11] Yuki Ono, Eduard Trulls, Pascal Fua, and Kwang Moo Yi. LF-Net: Learning Local Features from Images. NIPS, 2018.

[12] Kwang Moo Yi, Yannick Verdie, Pascal Fua, and Vincent Lepetit. Learning to assign orientations to feature points. CVPR, 2016.

REFERENCES

[13] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. CVPR Workshop, 2017.

[14] Yannick Verdie, Kwang Moo Yi, Pascal Fua, and Vincent Lepetit. Tilde: a temporally invariant learned detector. CVPR, 2015.

[15] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. TPAMI, 2005.

[16] Tinne Tuytelaars and Krystian Mikolajczyk. Local invariant feature detectors: a survey. Foundations and Trends in Computer Graphics and Vision, 2008.

[17] Chris Harris and Mike Stephens. A combined corner and edge detector. Alvey Vision Conference, 1988.

[18] Paul Beaudet. Rotationally invariant image operators. ICPR, 1978.

[19] Krystian Mikolajczyk, Tinne Tuytelaars, Cordelia Schmid, Andrew Zisserman, Jiri Matas, Frederik Schaffalitzky, Timor Kadir, and Luc Van Gool. A comparison of affine region detectors. IJCV, 2005.

[20] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). Computer Vision and Image Understanding, 2008.

[21] Pablo Fernandez Alcantarilla, Adrien Bartoli, and Andrew J. ´Davison. Kaze features. ECCV, 2012.

[22] Pablo Fernandez Alcantarilla, Jes ´ us Nuevo, and Adrien Bar- ´ toli. Fast explicit diffusion for accelerated features in nonlinear scale spaces. BMVC, 2013.

[23] Jiri Matas, Chum Ondrej, Urban Martin, and Pajdla Toms. Robust wide-baseline stereo from maximally stable extremal regions. Image and Vision Computing, 2004.

[24] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. ECCV, 2006.

REFERENCES

[25] Edward Rosten, Reid Porter, and Tom Drummond. Faster and better: A machine learning approach to corner detection. TPAMI, 2010.

[26] Stefan Leutenegger, Chli Margarita, and Siegwart Roland. Brisk: Binary robust invariant scalable keypoints. ICCV, 2011.

[27] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. ICCV, 2011.

[28] Nikolay Savinov, Akihito Seki, Lubor Ladicky, Torsten Sattler, and Marc Pollefeys. Quad-networks: unsupervised learning to rank for interest point detection. CVPR, 2017.

[29] Georgios Georgakis, Srikrishna Karanam, Ziyan Wu, Jan Ernst, and Jana Kosecka. End-to-end learning of keypoint detector and descriptor for pose invariant 3d matching. CVPR, 2018.

[30] Wilfried Hartmann, Michal Havlena, and Konrad Schindler. Predicting matchability. CVPR, 2014.

[31] Kwang Moo Yi, Eduard Trulls, Yuki Ono, Vincent Lepetit, Mathieu Salzmann, and Pascal Fua. Learning to find good correspondences. CVPR, 2018.

[32] Dmytro Mishkin, Filip Radenovic, and Jiri Matas. Repeatability is not enough: Learning affine regions via discriminability. ECCV, 2018.

[33] Luc Florack, Bart Ter Haar Romeny, Max Viergever, and Jan Koenderink. The gaussian scale-space paradigm and the multiscale local jet. IJCV, 2002.

[34] Krystian Mikolajczyk and Cordelia Schmid. Indexing based on scale invariant interest points. ICCV, 2001.

[35] Supasorn Suwajanakorn, Noah Snavely, Jonathan Tompson, and Mohammad Norouzi. Discovery of latent 3d keypoints via end-to-end geometric reasoning. NIPS, 2018.

[36] Jingming Dong and Stefano Soatto. Domain-size pooling in local descriptors: Dsp-sift. CVPR, 2017.

[37] Stepan Obdrzalek and Jiri Matas. Object recognition using local affine frames on distinguished regions. BMVC, 2002.

[38] Anastasiya Mishchuk, Dmytro Mishkin, Filip Radenovic, and Jiri Matas. Working hard to know your neighbor’s margins: Local descriptor learning loss. NIPS, 2017.

HOPE YOU ENJOYED

GTIHUB

SEMNAN-UNIVERSITY-AI/KEY.NET

TEAM

AMIR SHOKRI

Farshad Asgharzade

Alireza Gholamnia

EMAIL

AMIRSH.NLL@GMAIL.COM

Farshad_asgharzade@hotmail.com

gholamniareza@gmail.com