dacl10k: Benchmark for Semantic Bridge Damage Segmentation

Johannes Flotzinger Philipp J. Rösch Thomas Braml
University of the Bundeswehr Munich
{johannes.flotzinger, philipp.roesch, thomas.braml}@unibw.de

Abstract

Reliably identifying reinforced concrete defects (RCDs) plays a crucial role in assessing the structural integrity, traffic safety, and long-term durability of concrete bridges, which represent the most common bridge type worldwide. Nevertheless, available datasets for the recognition of RCDs are small in terms of size and class variety, which questions their usability in real-world scenarios and their role as a benchmark. Our contribution to this problem is "dacl10k", an exceptionally diverse RCD dataset for multi-label semantic segmentation comprising 9,920 images deriving from real-world bridge inspections. dacl10k distinguishes 12 damage classes as well as 6 bridge components that play a key role in the building assessment and recommending actions, such as restoration works, traffic load limitations or bridge closures. In addition, we examine baseline models for dacl10k which are subsequently evaluated. The best model achieves a mean intersection-over-union of 0.42 on the test set. dacl10k, along with our baselines, will be openly accessible to researchers and practitioners, representing the currently biggest dataset regarding number of images and class diversity for semantic segmentation in the bridge inspection domain.

1. Introduction

Bridges are an essential component of the infrastructure worldwide. They are exposed to many impacts causing damage, such as high traffic loads, extreme weather events, sea salt in coastal areas or treatment with deicing chemicals in cold regions. Due to their age, especially bridges in countries with an economic upswing between the 1950s and 1980s show an increased occurrence of damage by now. These defects are monitored within the scope of bridge inspections aiming to assess the condition of buildings in order to find the ideal timing for rehabilitation steps but also to take immediate actions, *e.g.*, limiting heavy traffic or closing a bridge. Such inspections are usually carried out "analogously" by professionally trained civil engineers who visually examine the complete surface of the bridge while taking photos of the defects, and documenting damage class, measurements and location on a 2D sketch [1,3,15]. If these examinations, specifically of buildings in a critical state, could take place more frequently, several bridges may be operated longer and without significant restrictions affecting rail passengers, car drivers and logistics. However, authorities often fail to keep up with the necessary inspection and restoration intervals due to staff shortages and budget limitations, but also because of the commonly applied, time-consuming, analogue inspection process used in practice. In many countries, this leads to a steadily growing stock of built structures in poor condition. In governmental reports, the current state of bridges is described as "1 in 3 U.S. bridges needs repair or replacement ..." [2] or "... at least 25,000 road bridges [in France] are in poor structural condition ..." [5]. This underlines the demand of a more efficient examination pipeline with respect to cost and time [5, 16, 17, 31]. The greatest potential for improvement during the bridge inspection process – irrespective of the used device (unmanned aerial vehicle (UAV) [19], smartphone [45], augmented reality displays [36]) – is yielded by automating the defect recognition which is crucial to a final assessment in order to determine actions to be taken. The inspection framework that makes use of automated defect recognition is called "digitized inspection" (DI). DIs aim for a detailed bridge assessment according to existing country-specific guidelines where the automatized documentation of damage allows reliably classifying, measuring and localizing each existing defect on a given building. Within the scope of DIs, inspectors are strongly supported. The inspections become more efficient and engineers can focus more on the evaluation as well as the context in which defects appear.

The field of damage recognition on built structures is still unexplored. In contrast to the fields of autonomous driving [9,12,18,34,44] or medicine [29,32,35], semantic segmentation benchmarks for damage recognition are rare. To the best of our knowledge, only two relevant benchmarks in the domain of reinforced concrete defects (RCDs) exist: CrackSeg9k [26] and S2DS [7]. CrackSeg9k is a collection of various image datasets showing cracked and uncracked surfaces of multiple building materials. However, it aims to

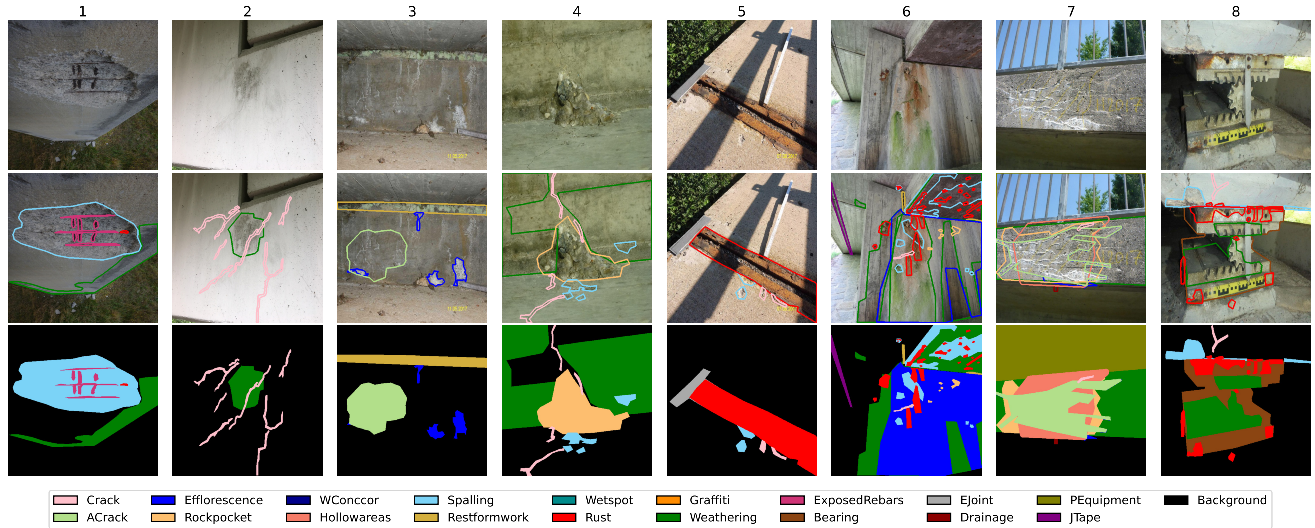

Figure 1. Example annotations from dacl10k. Top row: original image. Middle row: polygonal annotations. Bottom row: stacked masks. The following classes are abbreviated: *Alligator Crack* (ACrack), *Washouts/Concrete corrosion* (WConccor), *Expansion Joint* (EJoint), *Protective Equipment* (PEquipment) and *Joint Tape* (JTape). From left to right, the images display the individual classes: 1. *Weathering, Spalling, Exposed Rebars, Rust*; 2. *Weathering, Crack*; 3. *Alligator Crack, Restformwork, Efflorescence*; 4. *Weathering, Crack, Spalling, Rockpocket*; 5. *Crack, Rust, Expansion Joint, Spalling*; 6. *Weathering, Rockpocket, Spalling, Efflorescence, Crack, Rust, Restformwork, Joint Tape*; 7. *Weathering, Protective Equipment, Rockpocket, Efflorescence, Crack, Hollowareas, Alligator Crack, Drainage*; 8. *Weathering, Spalling, Crack, Rust, Bearing*.

segment one damage type, whereby, at least 9 types must be recognized in order to be used in practice [1, 3, 4, 15]. S2DS is the first multi-class semantic segmentation dataset in RCD domain which includes 743 samples differentiating between five common defects occurring on concrete bridges and control points for georeferencing. Thus, S2DS represents a small variety and complexity with respect to real-world scenarios, with labels assigned to each pixel in a manually exclusive way.

In conclusion, the significant deterioration of bridges worldwide as well as the lack of visual data for effectively monitoring their defects in a digitized manner emphasize the urgent need for establishing a benchmark in the bridge inspection domain.

We take the problem of semantic segmentation of bridge defects out of the niche by introducing *dacl10k*, the biggest real-world inspection dataset for multi-label semantic segmentation making it possible to perform damage classification, measurement and localization on a pixel-level. Thereby, we enable recognizing 12 frequently occurring defects on reinforced concrete bridges (*e.g.*, *Crack*, *Spalling*, *Efflorescence*) and 6 important building parts (*e.g.*, exposed reinforcement bar *Exposed Rebar*, *Bearing*, *Expansion Joint*, *Protective Equipment*). All these classes play an important role for determining the building's structural integrity, traffic safety and durability. dacl10k includes 9,920 images from more than 100 different bridges, specifically designed for practical use, as it comprises all visually unique damage types defined by bridge inspection standards. dacl10k surpasses previous work significantly in terms of its scale, class variety, and the complex nature of its captured scenes. Besides, we provide essential background knowledge from the civil engineering perspective, which is important for a deeper understanding of the RCD domain. In addition, we supply strong baselines to benchmark against. In our model analysis, two semantic segmentation architectures in combination with three encoders are examined. The dacl10k dataset, and according baselines, will be publicly released, fostering research within the field of damage recognition on concrete structures using computer vision.

2. Related datasets and baselines

Within the last six years, major contributions in the field of damage classification on built structures have been made through the introduction of datasets for binary classification [14, 23, 27, 43], multi-class classification [8, 22], multi-label classification [30], object detection [30], and semantic segmentation [6, 7, 26]. In the following, we discuss datasets for the last three named tasks. Examples for the subsequently named damage types can be obtained from Figure 1.

Mundt *et al*. [30] developed CODEBRIM which is currently the biggest and most realistic dataset for the multi-label classification of RCDs. They differ between the damage types: crack, spallation, exposed reinforcement bar, efflorescence, corrosion and background. The unbalanced version of CODEBRIM comprises 7,729 patches of defect images gathered from 30 bridges, chosen based on varying levels of deterioration, defect size, severity, and surface appearance. The images were acquired under changing weather conditions using multiple cameras at varying scales, with high-resolution. A subset of the data was acquired using an UAV, due to the inaccessibility of defects at high locations. Their annotation process was structured as follows: (*i*) selecting bounding boxes (patches) enclosing defects, (*ii*) iterating over the bounding boxes for each damage class and label accordingly, and finally (*iii*) sampling patches of healthy concrete surfaces as well as irrelevant content (background).

For solving the task of binary crack segmentation, Kulkarni *et al*. [26] combine previously available datasets, inter alia, from the RCD domain. They compile a semantic segmentation dataset, called CrackSeg9k, with 9,255 images of cracks from ten sub datasets on different surfaces. Before unifying the datasets, their individual problems (e.g. noise and distortion) are addressed by applying image processing. In addition, they provide baselines where the best model, based on DeepLabv3 [11], achieves 77% mean intersection-over-union (IoU).

Benz & Rodehorst [7] introduced the Structural Defect Dataset (S2DS) which is the first RCD dataset enabling semantic segmentation of multiple damage types, such as crack, spalling, corrosion, efflorescence, vegetation, and control point which is used for georeferencing. The dataset consists of 743 patches of size 1024x1024 pixel extracted from 8,435 images taken during structural inspections. They used DSLR cameras, mobile phones, and UAVs for acquiring the data. The labeling was executed by one trained computer scientist and had a high level of fineness. Their best model, based on hierarchical multi-scale attention [39], achieves a mean IoU of 92%, at joint scales of 0.25, 0.5, and 1.0.

3. dacl10k dataset

dacl10k is the first large-scale dataset for semantic bridge damage segmentation, comprising 9,920 annotated images from real-world inspections. During its creation, our primary objective was to develop a dataset that enables the training of models which later support the inspector during damage recognition and documentation to a maximum. Hence, we analyzed several guidelines determining the level of detail of structural inspections [1, 3, 4, 15], specifically the visually recognizable defects which must be collected in order to produce a legal bridge assessment. We listed all defects defined by the guidelines accordingly and crossed out the ones that are doppelgangers with respect to visual appearance. Finally, this resulted in the underlying class variety of dacl10k. In the following, we discuss dacl10k's data acquisition, classes, statistics and a comparison to related open-source data.

3.1. Data acquisition

Approximately one half of the images originate from databases of engineering offices, while the other half was provided by local authorities from Germany. The images were taken between 2000 and 2020. Both data sources supplied highly heterogeneous images regarding camera type, pose, lighting condition, and resolution. However, models performing well on dacl10k, most likely generalize well in real-world scenarios.

3.2. Damage types and annotation

The 18 classes considered within dacl10k are separated into three groups: concrete defects, general defects and objects. The class names are shown in the first column of Table 1. The concrete defects appear only on building parts made of (reinforced) concrete, while general defects may be present on all materials (*e.g*., concrete or steel). The only defect within dacl10k that is not visually recognizable, per se, is *Hollowareas*. This damage is usually identified by hammering on the concrete surface, thus, it can only be detected acoustically (not visually) but, as it is bordered with chalk during hands-on inspections, we annotated its markings. The objects group includes all components of a bridge that are not made of concrete, such as *Joint Tapes*, Railings or impact attenuation devices (*Protective Equipment*). The objects often show defects such as geometrical irregularities or deficits in structural capacity. Geometrical irregularities can arise from wrong distances between the railing rods or if the railing height is less than the minimum according to the given national standard. These visually challenging recognizable issues are not part of the dataset. We provide a detailed overview, descriptions and examples of the defect types and objects in the supplementary material.

According to the definition of Cordts *et al*. [12] our dataset comprises coarse pixel-level annotations. We border each defect and object on a given image with one polygon (shape) and assign its label. Furthermore, we include polygons of the same class that overlap with each other in one shape. With respect to inspection standards and application, for all the defect classes, it is not important to differentiate between instances of a given class. Instead, their size and localization on a class-basis is important. Thereby, we utilize the open-source labeling tool LabelMe [33, 40]. Example annotations are shown in Figure 1. It often appears that shapes of different damage or object classes overlap with each other, *e.g*., *Spalling* with *Exposed Rebars* covered by *Rust* (see example 1 in Figure 1). Consequently, the underlying task can be described as multi-label semantic segmentation because one pixel can be part of multiple defects and objects. In other words, the labels are not assigned mutually exclusive to the pixels.

The two main components during labeling are the class guideline and the annotation guideline. The class guideline clearly defines the visual appearance, most commonly occurrence and cause of each defect, *e.g.*, *Efflorescences* (see example 7 in Figure 1):

- look like stalactites of white to yellowish or reddish color hanging from the bottom of building parts which may also appear to be printed on the buildings surface,
- often occur in wet (*Wetspot*) or weathered areas (*Weathering*) of the building and in combination with *Crack* and/or *Rust*,
- result from the dissolving of salts from the concrete which consequently carbonate.

Thus, the annotator – independently from his domain expertise – can understand the color, shape and texture of a given class. Furthermore, flags are defined to mark images with personal data (faces, license plates) or images of bad quality. In addition, the provided annotation guideline describes the fineness after which the polygon points shall be set. The goal is to have consistent class and object-distance-dependent density of points over the whole dataset.

Our labeling process consisted of two consecutive steps: Annotation of data received from engineering offices by civil engineering students (in-house) and annotation data from the authorities by an external annotation team, previously filtered for relevant image content. The students labeled approximately 7,000 images in accordance to our guidelines. Nearly 30% of the images had to be rejected due to flags indicating bad quality (blurring, overexposure) or personal data. The pipeline during in-house labeling was separated into three parts. The first part consisted of the regular annotation which comprises annotating a batch of 100 images, getting feedback by a domain expert and correcting the failures accordingly. The second part included an extensive analysis of the dataset to find structural failures in the annotations. Thirdly, the dataset was divided into subtasks with respect to the failures most commonly made. Then, one student corrected each failure type consecutively.

The quality assessment of the data annotated by the external team was divided into four quality checks for each data batch. In average, one batch included 250 images. Each check included one iteration over the annotated data by experts. Based on the analysis, the error rate was determined which is the ratio of false-labeled images and total amount of frames in the according batch. Starting with an error rate of 60%, the rate could be lowered to a final value of 1%.

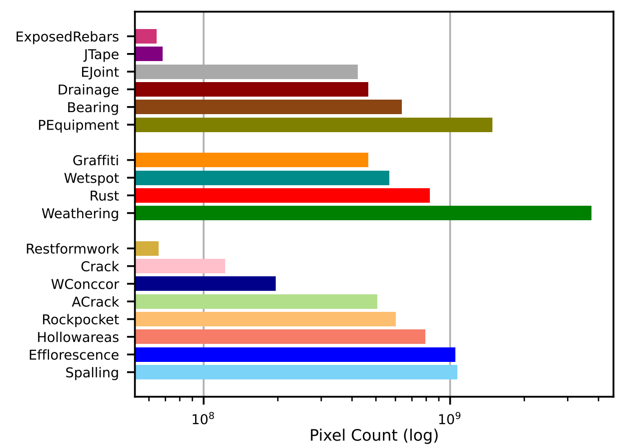

Figure 2. Pixel counts with respect to each class in dacl10k based on the original image sizes. The bars are arranged according to the group affiliation.

3.3. Statistical analysis

In the following, we discuss split-independently (*i*) the classwise pixel counts, (*ii*) averages and shares regarding the polygons and pixels as well as (*iii*) the split-dependent statistics, based on the original image resolutions.

Regarding the pixel counts over the whole dataset (see Figure 2), it can be stated that *Weathering*, followed by *Protective Equipment*, are the most dominant classes with nearly four and 1.5 billion pixels. Within the range of 0.1 billion and 1 billion pixels, the majority of defects and objects with respect to the number of pixels can be found. Clearly underrepresented are the defects *Restformwork* (66 million) and the objects *Joint Tape* (68 million) and *Exposed Rebars* (65 million).

The average image size is 1581 px in height and 1950 px in width. The mean image area is approximately 4 megapixels while the total pixel area in the dataset is approx. 43 billion px. Table 1 provides an overview of statistics describing the classwise density of polygons, size of polygons, density of pixels, share of polygons and share of pixels over the whole dataset. In average, 1.8 crack shapes are present on a crack image, whereby, one polygon includes 27,467 px. The average crack image shows approx. 50,000 px labeled as crack. With respect to the displayed shares, we observe that four out of 100 polygons are labeled as crack and 0.3% of the total pixel area received the label *Crack*. Furthermore, Table 1 reveals the cause of the overrepresented classes *Weathering* and *Protective Equipment*. They display a share regarding the number of polygons of 5.31% (top 20%) and 2.09% (exactly the median). This, in combination with the fact that an according image shows 900,000 px or rather 786,000 px of that class, leads to their dominant role. The overrepresentation of *Weathering* can be

Class	#polyg./ image	#pixels/ polyg.	#pixels/ image	%polyg.	%pixels
Crack	1.81	27,467	49,605	4.02	0.30
ACrack	1.12	950,694	1,064,777	0.48	1.25
Efflorescence	2.29	208,565	478,395	4.54	2.59
Rockpocket	4.75	50,564	240,079	10.74	1.48
WashoutsC.	1.34	807,721	1,079,936	0.22	0.48
Hollowareas	1.21	415,627	504,482	1.72	1.96
Spalling	2.68	83,298	223,185	11.60	2.64
Restformw.	1.19	50,170	59,757	1.18	0.16
Wetspot	1.48	271,408	400,778	1.89	1.40
Rust	3.62	46,680	168,997	16.01	2.04
Graffiti	2.29	172,917	395,596	2.44	1.15
Weathering	1.41	639,776	903,974	5.31	9.28
ExposedR.	2.25	25,770	58,034	2.26	0.16
Bearing	1.45	421,784	612,581	1.37	1.57
EJoint	1.12	700,054	783,023	0.55	1.05
Drainage	1.37	230,815	316,099	1.83	1.15
PEquipment	1.22	641,715	785,601	2.09	3.67
JTape	1.19	49,401	58,569	1.25	0.17
Background	3.47	871,631	3,020,476	30.51	72.69

Table 1. Overall statistics of the dataset regarding average number of polygons per image, number of pixels per polygon, number of pixels per image, share of polygons and share of pixels. Midrules separate the classes according to their group affiliation.

more fatal than the one of *Protective Equipment* with respect to the model performance. This is due to the fact that the features (shape and texture) of *Weathering* are similar to the ones from *Wetspot*. Both are of round shape and represented by a darker area surrounded by a brighter "rest". They vary slightly with regards to their texture. *Weathering* is more noisy and more matt than *Wetspot* which is smooth and sometimes mirroring. In addition, *Wetspot* and *Weathering* often overlap which makes it difficult to distinguish between them from the model's perspective. The features of the object *Protective Equipment*, in contrast, are unique and therefore shouldn't interfere with other classes during learning. The lack of pixels representing *Restformwork*, *Exposed Rebars* and *Joint Tape*, as mentioned before, originates from their relatively rare occurrences but mostly from their small shapes. In average, polygons bordering *Restformwork* or *Joint Tape* have a size of approx. 50,000 px while polygons labeled as *Exposed Rebars* include 26,000 px. The average size of their polygons is equal to or less than the lower quartile (50,367 px). Additionally, *Exposed Rebars* shows the smallest average polygon size of all classes.

To ensure similar data distributions (regarding damage and object classes) in each split, we create data partitions according to image similarity. For that, we employ K-means clustering [28] with 20 clusters. Then, we proportionally draw samples from each cluster for the train (70%), valid (10%), testdev (10%), and testpriv (10%) set with respect to the number of polygons and images (see Table 2). The testdev and testpriv split are summarized within the test split.

The Table presents the classwise statistics of dacl10k separated into the given splits. Again, taking the *Crack* class as an example, the aforementioned proportions can be observed in terms of number of pixels with a number of 89,316,599 px (73%) in the train and approx. 11,000 px (9%) in the validation split. Compared to the validation split, the values of the test split are approx. twice as high. For the number of polygons and images, Table 2 displays the same targeted proportions as for the pixel count.

3.4. Comparison to other datasets

Compared to CrackSeg9k [26] and S2DS [7], the crack annotations of dacl10k are coarser. CrackSeg9k is a collection of multiple available binary crack segmentation datasets where each was acquired in a standardized setting with respect to camera pose, lighting condition and hardware. S2DS is a single-label semantic segmentation dataset that includes images of RCDs (and control points) captured during real structural inspections, where the fineness of crack annotations is also high. According to the inspection guidelines, the recognition of cracks requires the highest degree of accuracy because their width plays an important role during their assessment. *E.g.*, in Germany the minimum crack width, which must be documented, is 0.2mm [4, 15]. Consequently, for the damage type *Crack*, finer annotations are definitely useful when it comes to practical use.

To sum up, both related datasets show a higher level of detail regarding crack annotations than dacl10k but are less diverse with respect to class variety and real-world scenarios. CrackSeg9k enables the training of models for binary crack segmentation only. The annotations of S2DS provide single-target information at most, meaning that one pixel can only belong to one class. In addition, S2DS consists of a relatively small number of images. However, we focus on the multi-label semantic segmentation of all visually unique defects and objects on concrete bridges. Thereby, we take into account the frequently occurring case that in real-world scenarios multiple defects overlap. Regarding the crack annotations, it can be stated that a pixel-accurate classification of our crack data may be possible by applying methods from the field of weakly-labeled data [21]. In the CityScapes dataset [12], for example, the majority of the samples were coarsely annotated (20,000 images) with the intention to foster research specifically in this field. With regards to applications within the framework of existing standards, all other classes in our dataset do not require a finer annotation and prediction, respectively. *E.g.*, during currently practiced analogue inspections, the diameter and corresponding area of a *Spalling* is usually measured in a rough manner with a folding rule which is sufficient for its assessment.

Class	Train			Valid			Test		
	#pixels	#polyg.	#images	#pixels	#polyg.	#images	#pixels	#polyg.	#images
Crack	89,316,599	3,092	1,720	11,462,336	457	254	21,199,921	892	485
ACrack	379,337,903	378	336	33,145,894	48	42	93,285,498	106	97
Efflorescence	773,838,619	3,378	1,523	69,295,099	502	206	204,072,743	1,141	460
Rockpocket	416,831,564	8,241	1,712	51,701,480	1,207	259	131,663,248	2,422	529
WConccor	153,351,739	176	133	7,781,177	23	15	34,335,547	43	33
Hollowareas	589,108,842	1,327	1,100	68,277,162	193	155	133,137,140	382	312
Spalling	754,421,298	8,638	3,289	94,740,019	1,444	485	218,557,811	2,736	1,010
Restformwork	42,413,460	841	716	6,143,333	164	132	17,116,171	304	251
Wetspot	402,563,879	1,436	972	47,002,103	216	144	117,133,455	436	298
Rust	592,984,180	12,272	3,451	79,135,671	1,801	465	153,938,160	3,623	972
Graffiti	308,779,763	1,866	797	71,491,732	317	146	85,741,013	512	235
Weathering	2,563,181,494	4,056	2,830	367,950,798	572	407	823,072,419	1,240	916
ExposedRebars	44,829,810	1,720	773	3,824,934	244	104	15,821,469	538	234
Bearing	431,016,513	1,039	731	89,236,423	160	105	116,219,161	310	203
EJoint	335,469,869	446	396	21,145,699	56	51	66,216,825	102	93
Drainage	368,660,666	1,393	1,030	35,297,153	244	151	62,288,022	383	294
PEquipment	1,070,432,844	1,616	1,320	103,082,479	211	175	312,055,649	488	396
JTape	47,582,309	911	772	8,569,024	152	128	12,022,699	317	264
Background	21,196,542,072	23,347	6,801	2,681,148,040	3,279	962	5,514,566,563	7,095	1,968

Table 2. Splitwise statistics regarding the number of pixels, polygons and images.

4. Baselines

In the following, we describe the development of the baseline models and their test results. In order to evaluate the baselines and to demonstrate challenges with respect to the annotation of the underlying data, we conduct an "Engineer versus Machine" (EvsM) comparison. Finally, we discuss the results incorporating the findings regarding dacl10k, the baselines and the EvsM comparison.

4.1. Implementation details

For the development of the baselines we analyze two different CNN-based semantic segmentation architectures with three different encoders, and one Transformer-based model. We used DeepLabV3+ [10,11] and Feature Pyramid Network (FPN) [25] as CNN baseline architectures which represent powerful models often used in research and industry. As encoders we utilize MobileNetV3-Large [20] (3M parameters), EfficientNet-B2 [38] (7M parameters) and EfficientNet-B4 (17M parameters). For each of these base models we also investigate a separate model using an auxiliary loss for multi-label classification. The auxiliary classification head placed after the encoder consists of an average global pooling layer, followed by a dropout and linear layer. The weights of the model are updated based on a combined weighted loss including the mask and auxiliary loss. The total loss $\mathcal{L}_{total}$ is computed as follows:

$$\mathcal{L}_{total} = \mathcal{L}_{mask} + 0.1 \times \mathcal{L}_{aux} \tag{1}$$

where $\mathcal{L}_{mask}$ and $\mathcal{L}_{aux}$ are based on Dice [37] and Cross Entropy loss respectively.

To meet current developments in Transformer-based models, we trained a SegFormer model [42]. Since auxiliary loss is unusual for this network, only Dice loss is utilized. Adam optimizer [24] with four different learning rates ($5e^{-3}$, $1e^{-3}$, $5e^{-4}$, $1e^{-4}$) is applied whereof the model with the best loss based on the validation split is reported. We make use of a cosine learning rate scheduler with a warm-up phase over two epochs and train each model for 30 epochs. All models are initialized with ImageNet weights [13, 41].

The images and annotations are resized to a resolution of 512×512 before being fed to the network. Thereby, each defect and object class is considered with a separate binary mask, leading to a total number of 18 channels. All following results are reported on the same resolution, enabling an evaluation that is not focused towards large images.

4.2. Baseline results

In order to find the best performing model on dacl10k, we compare the mean IoU of seven different models (see Table 3). The best network using the DeepLabV3+ architecture includes an EfficientNet-B4 backbone without considering auxiliary loss. It achieves a mean IoU of 0.411. The highest mean IoU is obtained at a value of 0.414 by the model consisting of an EfficientNet-B4 encoder and FPN architecture while taking the auxiliary loss into account. For both, FPN and DeepLabV3+, it can be observed that the more parameters the encoder has, the higher is the

Aux	DeepLabv3+			FPN			SegFor
	MN	EN-B2	EN-B4	MN	EN-B2	EN-B4	
–	0.320	0.360	0.411	0.376	0.384	0.364	0.400
✓	0.329	0.400	0.409	0.378	0.395	**0.414**	–

Table 3. Mean IoU on valid split for both architectures and three encoders (MobileNetV3-Large, EfficientNet-B2 and EfficientNet-B4) with and without auxiliary loss, and the SegFormer model.

reached mean IoU. The SegFormer model achieves a mean IoU of 0.400 which is less than the best model based on DeepLabV3+ or FPN.

Table 4 displays the classwise IoUs and the mean IoU of the best model, described in the preceded paragraph. We report results on the validation and test split (see Table 2). Due to the small differences between the metrics over the splits, it can be stated that the data, with respect to complexity, is evenly distributed. In the following, the results on the test split are documented. The lowest IoU is obtained for the defect *Washouts/Concrete Corrosion* with a value of 0.121. This class is not underrepresented (see Figure 2). We observe that its texture and shape (features) are very familiar to the ones from *Rockpocket* and *Spalling* which both are strongly represented. Other classes for which a low IoU is reported are *Wetspot*, *Restformwork*, *Crack* and *Rockpocket*. The possible reason for the bad performance on *Wetspot* is explained in Section 3.3. *Restformwork* has many different visual appearances, which the model probably fails to summarize within one class. The *Crack* class, in average, is represented by the least number of pixels per image, and it has the smallest polygons, after *Exposed Rebars*, as they are elongated and very narrow (see Table 1). Thus, the IoU is a challenging metric for this defect, as the false-negative segmentation of classes represented by more pixel area is less penalized. On *Rockpocket* a low IoU is obtained because of the aforementioned similar-looking defects. The best results can be reported for the objects *Protective Equipment* (0.715) and *Bearing* (0.564) while the best general defect is *Graffiti* (0.623) and with respct to concrete defects, a good IoU can be observed for *Hollowareas* (0.555) and *Alligator Crack* (0.482). *Graffiti* and *Protective Equipment* are the two most individual classes regarding their visual appearance, additionally, they are well represented (see Figure 2) which explains their high IoU compared to the rest of the classes. Summarizing, the best model achieves a mean IoU of 0.424. A more detailed analysis of the problematic classes is provided within the supplementary material.

5. Engineer vs. Machine

Figure 3 displays the EvsM comparison, which enables a qualitative validation of segmentations from the best model (see Section 4.2) by comparing the annotations of civil engineers with the network's predictions. Therefore, we asked the five experts, two of whom perform bridge inspections frequently, to annotate four representative samples drawn from dacl10k's testdev split. Two samples are considered easy and the others difficult to evaluate because of their low resolution, diversity regarding the lighting condition, camera pose as well as defect types and objects. After an introduction to the class guideline, the experts were instructed to annotate the images with the same quality they would expect an application to highlight defects during a close-up or hands-on bridge inspection. Compared to other inspection types, hands-on inspections require the highest quality with respect to defect classification, measurement and localization, which also includes the detection of *Hollowareas* by hammering the concrete surface. Regarding the differences among engineers, it can be stated that with increasing complexity of the samples also the variance of the chosen labels rises. Engineer 5 annotated at the highest quality level. Thus, especially his annotation is used as a qualitative benchmark for our baselines prediction. For the first two samples, which contain fewer classes compared to the last two, the prediction of the machine is better than the average annotation of the five engineers. Only in the second image the small cavities that are part of the defect *Rockpocket* and the *Protective Equipment*, which is overexposed, are not recognized. The last two images reveal the limits of our baseline. For the third sample the classes *Wetspot*, *Rust*

Class	valid	test
Crack	0.288	0.286
ACrack	0.473	0.482
Efflorescence	0.338	0.415
Rockpocket	0.267	0.294
WConccor	0.085	0.121
Hollowareas	0.536	0.555
Spalling	0.374	0.406
Restformwork	0.336	0.285
Wetspot	0.232	0.243
Rust	0.414	0.450
Graffiti	0.586	0.623
Weathering	0.423	0.395
ExposedRebars	0.393	0.358
Bearing	0.676	0.564
EJoint	0.474	0.524
Drainage	0.521	0.563
PEquipment	0.675	0.715
JTape	0.362	0.362
Mean	0.414	0.424

Table 4. Classwise and mean IoU of the best model (FPN with EfficientNet-B4 and auxiliary loss) on the validation and test split.

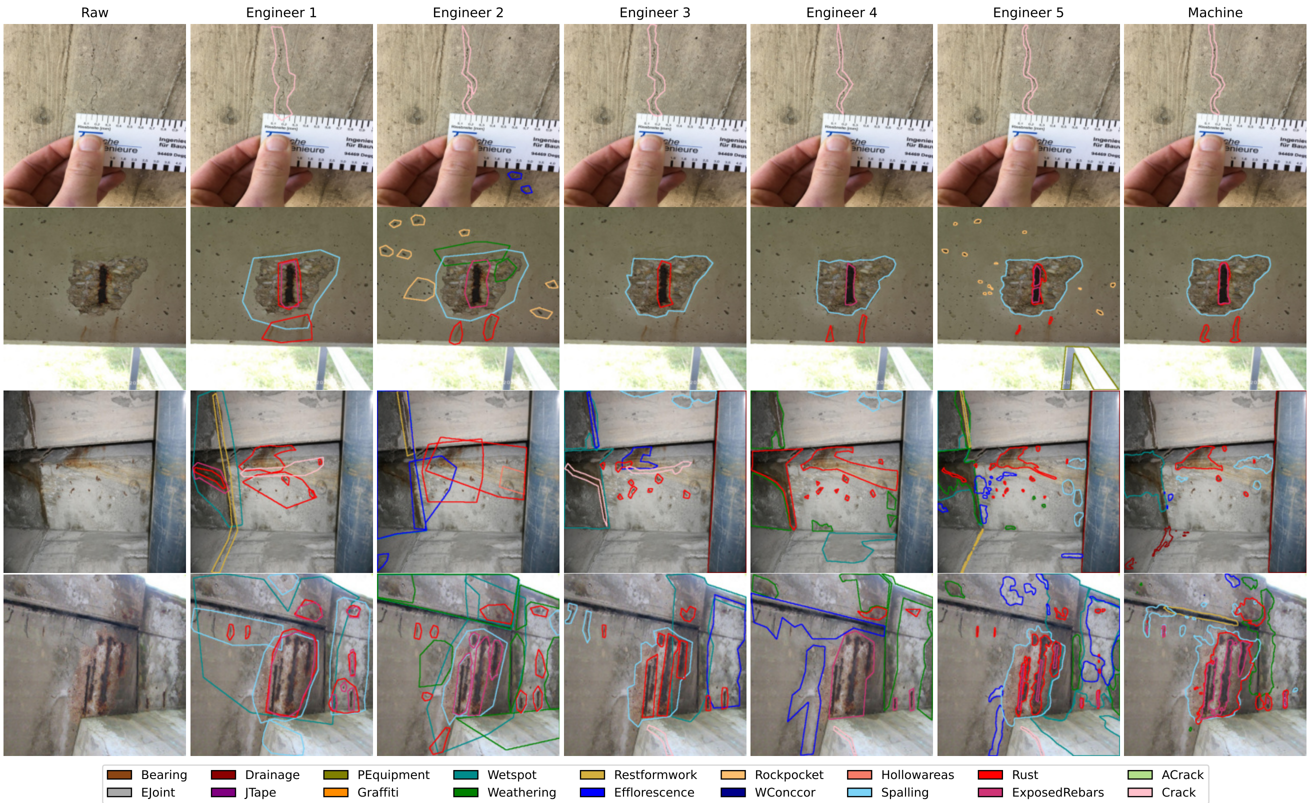

Figure 3. Qualitative evaluation of the best model on four samples (Raw) from the testdev split by comparing annotations of five civil engineers (Engineer 1-5) with the best model's predictions (Machine). From top to bottom, the samples show: *Crack*; a few tiny cavities which are considered as *Rockpockets* and *Spalling, Exposed Rebars, Rust*; *Wetspot, Weathering, Efflorescence, Crack, Spalling, Exposed Rebars and Rust*; *Wetspot, Weathering, Efflorescence, Restformwork, Drainage, Rust, Spalling*.

and *Drainage* are well predicted, whereas, areas of *Efflorescence* and weak *Spallings* are not segmented. *Weathering* is not present at all in the model's prediction. On the last image the model doesn't predict the defects *Crack* at the center bottom and the *Wetspot* which is located mainly on the wall. Furthermore, the model classifies *Restformwork* which is not present on the image, while *Spalling* and *Exposed Rebars* are correctly recognized. The remaining classes are partially classified.

6. Discussion

We have introduced the first large-scale dataset and corresponding baselines for multi-label semantic segmentation in the bridge inspection domain. It's based on real-world data, with a label distribution deriving from the visually distinguishable classes of multiple country-specific inspection standards. It is important to note that the concrete and general defect group labels are not restricted to bridges, as they can occur on any building made of (reinforced) concrete. The evaluation of the baselines, especially the EvsM comparison, generally shows a good performance. However, limitations can be observed for classes with minimal feature differences between each other, *e.g.*, *Weathering* and *Wetspot* or *Washouts/Concrete Corrosion*, *Rockpocket* and *Spalling*. In addition, the datatset is highly unbalanced, leading to biases towards the overrepresented classes.

We are confident that a more sophisticated search for architectures and hyperparameters as well as data augmentation methods will lead to substantial improvements. Additionally, with respect to finer *Crack* segmentations, approaches from the field of weakly-labeled data may better satisfy the geometrical accuracy requirements of this defect type. Overall, we believe that due to its size and diversity, dacl10k makes an important contribution to the field of automated structural inspection.

Acknowledgements. We thank Wolfgang Kusterle, Helmut Mayer and Johannes Kreutz for their insights and suggestions. We are most grateful to the engineering offices and authorities for providing the inspection images. This project has received funding from the Bavarian Ministry of Economic Affairs (MoBaP research project, IUK 639/003).

References

[1] AASHTO. *Manual for Bridge Evaluation, 3rd Edition*. American Association of State Highway and Transportation Officials, Washington, D.C., 2018. 1, 2, 3

[2] American Road & Transportation Builders Association (ARTBA). ARTBA bridge report, 2023. 1

[3] Austrian Research Association for Roads, Railways and Transport (RVS). Quality assurance for structural maintenance; Surveillance, checking and assessment of bridges and tunnels; Road bridges. Standard RVS 13.03.11, Vienna, AUT, 2011. 1, 2, 3

[4] Federal Highway Research Institute (BaSt). Guideline for the uniform acquisition, assessment, recording and evaluation of results of structural inspections (RI-EBW-PRÜF), Feb. 2017. 2, 3, 5

[5] Bruno M. Belin. Rapport d'information. *Session Ordinaire*, 669, 2022. 1

[6] Christian Benz, Paul Debus, Huy Khanh Ha, and Volker Rodehorst. Crack segmentation on uas-based imagery using transfer learning. In *2019 International Conference on Image and Vision Computing New Zealand (IVCNZ)*, pages 1–6, 2019. 2

[7] Christian Benz and Volker Rodehorst. Image-based detection of structural defects using hierarchical multi-scale attention. In *DAGM German Conference on Pattern Recognition (GCPR)*, pages 337–353. Springer, 2022. 1, 2, 3, 5

[8] Eric Bianchi and Matt Hebdon. Bearing Condition State Classification Dataset. University Libraries, Virginia Tech, Oct. 2021. 2

[9] Gabriel J. Brostow, Julien Fauqueur, and Roberto Cipolla. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, 30(2):88–97, 2009. Video-based Object and Event Analysis. 1

[10] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation, 2017. 6

[11] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation, 2018. 3, 6

[12] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1, 3, 5

[13] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 6

[14] Sattar Dorafshan, Robert J. Thomas, and Marc Maguire. SDNET2018: An annotated image dataset for non-contact concrete crack detection using deep convolutional neural networks. *Data in Brief*, 21:1664–1668, Dec. 2018. 2

[15] German Institute for Standardization Registered Association (DIN). Engineering structures in connection with roads - inspection and test. Standard DIN 1076:1999-11, Berlin, GER, 1999. 1, 2, 3, 5

[16] RAC Foundation. Substandard road bridges in Great Britain 2021, 2022. 1

[17] BaSt Bundesanstalt für Strassenwesen. Brückenstatistik, 2023. 1

[18] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 1

[19] Gi-Hun Gwon, Jin Hwan Lee, In-Ho Kim, and Hyung-Jo Jung. Cnn-based image quality classification considering quality degradation in bridge inspection using an unmanned aerial vehicle. *IEEE Access*, 11:22096–22113, 2023. 1

[20] A. Howard, M. Sandler, B. Chen, W. Wang, L. Chen, M. Tan, G. Chu, V. Vasudevan, Y. Zhu, R. Pang, H. Adam, and Q. Le. Searching for mobilenetv3. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1314–1324, Los Alamitos, CA, USA, nov 2019. IEEE Computer Society. 6

[21] Ronghang Hu, Piotr Dollár, Kaiming He, Trevor Darrell, and Ross Girshick. Learning to segment every thing. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4233–4241, 2018. 5

[22] Philipp Hüthwohl, Ruodan Lu, and Ioannis Brilakis. Multi-classifier for reinforced concrete bridge defects. *Automation in Construction*, 105, Sep 2019. 2

[23] Philipp Hüthwohl and Ioannis Brilakis. Detecting healthy concrete surfaces. *Adv. Eng. Inf.*, 37:150–162, Aug. 2018. 2

[24] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. 6

[25] A. Kirillov, R. Girshick, K. He, and P. Dollar. Panoptic feature pyramid networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6392–6401, Los Alamitos, CA, USA, jun 2019. IEEE Computer Society. 6

[26] Shreyas Kulkarni, Shreyas Singh, Dhananjay Balakrishnan, Siddharth Sharma, Saipraneeth Devunuri, and Sai Chowdeswara Rao Korlapati. Crackseg9k: A collection and benchmark for crack segmentation datasets and frameworks. In Leonid Karlinsky, Tomer Michaeli, and Ko Nishino, editors, *Computer Vision – ECCV 2022 Workshops*, pages 179–195, Cham, 2023. Springer Nature Switzerland. 1, 2, 3, 5

[27] Shengyuan Li and Xuefeng Zhao. Image-Based Concrete Crack Detection Using Convolutional Neural Network and Exhaustive Search Technique. *Advances in Civil Engineering*, 2019(Ml), 2019. 2

[28] S. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982. 5

[29] Bjoern H. Menze, Andras Jakab, Stefan Bauer, Jayashree Kalpathy-Cramer, Keyvan Farahani, Justin Kirby, Yuliya Burren, Nicole Porz, Johannes Slotboom, Roland Wiest, Levente Lanczi, Elizabeth Gerstner, Marc-André Weber, Tal Arbel, Brian B. Avants, Nicholas Ayache, Patricia Buendia, D. Louis Collins, Nicolas Cordier, Jason J. Corso, Antonio Criminisi, Tilak Das, Hervé Delingette, Çağatay Demiralp,

Christopher R. Durst, Michel Dojat, Senan Doyle, Joana Festa, Florence Forbes, Ezequiel Geremia, Ben Glocker, Polina Golland, Xiaotao Guo, Andac Hamamci, Khan M. Iftekharuddin, Raj Jena, Nigel M. John, Ender Konukoglu, Danial Lashkari, José António Mariz, Raphael Meier, Sérgio Pereira, Doina Precup, Stephen J. Price, Tammy Riklin Raviv, Syed M. S. Reza, Michael Ryan, Duygu Sarikaya, Lawrence Schwartz, Hoo-Chang Shin, Jamie Shotton, Carlos A. Silva, Nuno Sousa, Nagesh K. Subbanna, Gabor Szekely, Thomas J. Taylor, Owen M. Thomas, Nicholas J. Tustison, Gozde Unal, Flor Vasseur, Max Wintermark, Dong Hye Ye, Liang Zhao, Binsheng Zhao, Darko Zikic, Marcel Prastawa, Mauricio Reyes, and Koen Van Leemput. The multimodal brain tumor image segmentation benchmark (brats). *IEEE Transactions on Medical Imaging*, 34(10):1993–2024, 2015. 1

[30] Martin Mundt, Sagnik Majumder, Sreenivas Murali, Panagiotis Panetsos, and Visvanathan Ramesh. Meta-learning convolutional neural architectures for multi-target concrete defect classification with the concrete defect bridge image dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[31] American Society of Civil Engineers (ASCE). 2021 Infrastructure Report Card, 2021. 1

[32] Patrik F. Raudaschl, Paolo Zaffino, Gregory C. Sharp, Maria Francesca Spadea, Antong Chen, Benoit M. Dawant, Thomas Albrecht, Tobias Gass, Christoph Langguth, Marcel Lüthi, Florian Jung, Oliver Knapp, Stefan Wesarg, Richard Mannion-Haworth, Mike Bowes, Annaliese Ashman, Gwenael Guillard, Alan Brett, Graham Vincent, Mauricio Orbes-Arteaga, David Cárdenas-Peña, German Castellanos-Dominguez, Nava Aghdasi, Yangming Li, Angelique Berens, Kris Moe, Blake Hannaford, Rainer Schubert, and Karl D. Fritscher. Evaluation of segmentation methods on head and neck ct: Auto-segmentation challenge 2015. *Medical Physics*, 44(5):2020–2036, 2017. 1

[33] Bryan C. Russell, Antonio Torralba, Kevin P. Murphy, and William T. Freeman. LabelMe: A database and web-based tool for image annotation. *International Journal of Computer Vision*, 77(1-3):157–173, Oct. 2007. 3

[34] Timo Scharwächter, Markus Enzweiler, Uwe Franke, and Stefan Roth. Efficient multi-cue scene segmentation. In Joachim Weickert, Matthias Hein, and Bernt Schiele, editors, *Pattern Recognition*, pages 435–445, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. 1

[35] Amber L. Simpson, Michela Antonelli, Spyridon Bakas, Michel Bilello, Keyvan Farahani, Bram van Ginneken, Annette Kopp-Schneider, Bennett A. Landman, Geert Litjens, Bjoern Menze, Olaf Ronneberger, Ronald M. Summers, Patrick Bilic, Patrick F. Christ, Richard K. G. Do, Marc Gollub, Jennifer Golia-Pernicka, Stephan H. Heckers, William R. Jarnagin, Maureen K. McHugo, Sandy Napel, Eugene Vorontsov, Lena Maier-Hein, and M. Jorge Cardoso. A large annotated medical image dataset for the development and evaluation of segmentation algorithms, 2019. 1

[36] Alan Smith, Charlie Duff, Rodrigo Sarlo, and Joseph L. Gabbard. Wearable augmented reality interface design for bridge inspection. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pages 497–501, 2022. 1

[37] Carole H. Sudre, Wenqi Li, Tom Vercauteren, Sebastien Ourselin, and M. Jorge Cardoso. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pages 240–248. Springer International Publishing, 2017. 6

[38] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks, 2020. 6

[39] Andrew Tao, Karan Sapra, and Bryan Catanzaro. Hierarchical multi-scale attention for semantic segmentation, 2020. 3

[40] Kentaro Wada. labelme: Image polygonal annotation with python. https://github.com/wkentaro/labelme, 2018. 3

[41] Ross Wightman. Pytorch image models. https://github.com/rwightman/pytorch-image-models, 2019. 6

[42] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers, 2021. 6

[43] Hongyan Xu, Xiu Su, Huaiyuan Xu, and Haotian Li. Autonomous bridge crack detection using deep convolutional neural networks. In *3rd International Conference on Computer Engineering, Information Science & Application Technology (ICCIA 2019)*, pages 274–284. Atlantis Press, 2019. 2

[44] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2633–2642, 2020. 1

[45] Mahta Zakaria, Enes Karaaslan, and F. Necati Catbas. Advanced bridge visual inspection using real-time machine learning in edge devices. *Advances in Bridge Engineering*, 3(1), Dec. 2022. 1