Vol. 30, No. 1/4, 2021

Machine GRAPHICS & VISION

International Journal

Published by
The Institute of Information Technology
Warsaw University of Life Sciences – SGGW
Nowoursynowska 159, 02-776 Warsaw, Poland
in cooperation with
The Association for Image Processing, Poland – TPO

On the Use of Convolutional Neural Networks with Patterned Stride for Medical Image Analysis

Luiz Zaniolo, Oge Marques
Florida Atlantic University, Boca Raton, FL (USA)

Abstract. The use of deep learning techniques for early and accurate medical image diagnosis has grown significantly in recent years, with some encouraging results across many medical specialties, pathologies, and image types. One of the most popular deep neural network architectures is the convolutional neural network (CNN), widely used for medical image classification and segmentation, among other tasks. One of the configuration parameters of a CNN is called *stride* and it regulates how sparsely the image is sampled during the convolutional process. This paper explores the idea of applying a patterned stride strategy: pixels closer to the center are processed with a smaller stride concentrating the amount of information sampled, and pixels away from the center are processed with larger strides consequently making those areas to be sampled more sparsely. We apply this method to different medical image classification tasks and demonstrate experimentally how the proposed patterned stride mechanism outperforms a baseline solution with the same computational cost (processing and memory). We also discuss the relevance and potential future extensions of the proposed method.

Key words: convolutional neural networks, patterned stride, medical image classification, deep learning.

1. Introduction

The use of deep learning architectures for medical image analysis has experienced significant growth in recent years [12], with impressive success stories and claims of superhuman performance across many tasks, image modalities, and diseases.

Convolutional Neural Networks (CNNs) are the most popular architecture for medical image classification tasks. CNN architectures have an end-to-end structure, which learn high-level representations from raw data [11] without the need for pre-selecting features relevant to that type of data. Ever since their initial success on ImageNet Large-Scale Visual Recognition Challenge (LSVRC) 2012 [10], CNNs have been extended to specialized image classification tasks.

The performance of a CNN can be improved by fine-tuning some of its parameters (e.g., number of layers, type of layers, weights, biases) and hyperparameters (e.g., learning rate, number of epochs, loss function, activation functions). One of the CNN parameters that can be specified by the designer of the network is called *stride*: it specifies how the filters in a convolutional layer convolve around the input volume. Larger stride values mean that the network will perform more sparse convolutions – and consequently

a smaller number of them – throughout the processing of the image. Smaller stride values mean more concentrated sampling and consequently larger number of convolutions performed. The majority of existing CNNs implement a fixed value of stride, usually between 1 and 3. The minimum stride value of 1 means that all pixels are processed, which sets an upper bound on the computational cost of the convolution operations.

This paper explores the idea of changing the stride value in CNNs depending on the position of the pixel within the image: a smaller stride value is used when processing the center of the image, while a larger one is used for pixels close to the edges, according to a predefined sampling pattern.

Guo et al. [6] proposed another method where image classification benefits from using a patterned stride. However, their algorithm addresses the image classification problem in a different way, by judging the image complexity based on extracted features, and using that result to decide the stride value, suggesting smaller stride values for complex images and larger strides for simple ones.

The proposed method is experimentally evaluated on three medical image analysis tasks – (i) skin lesion classification, (ii) brain tumor detection, and (iii) image modality classification – and compared against a baseline fixed stride approach that requires similar computational power for both training and inference phases.

2. Materials and Methods

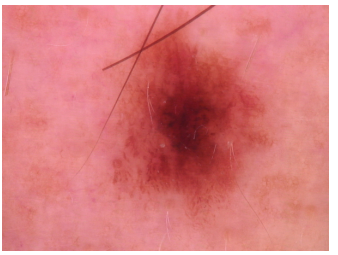
2.1. Datasets

In this study, we use three datasets: the HAM 10000 dataset used for skin lesion classification tasks; the Brain Tumor dataset, which contains images with healthy and unhealthy magnetic resonance imaging (MRI) brain slices; and the MedNIST dataset for image modality classification.

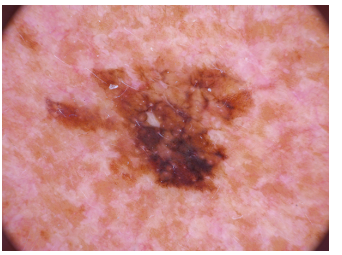
2.1.1. The HAM 10000 Dataset

The HAM10000 (*Human Against Machine with 10000 training images*) dataset [15] is a large collection of labeled multi-source dermatoscopic images in RGB color space, manually classified into one of seven different classes: *Melanocytic nevi*, *Melanoma*, *Benign keratosis-like lesions*, *Basal cell carcinoma*, *Actinic keratoses*, *Vascular lesions*, and *Dermatofibroma* (Figure 1).

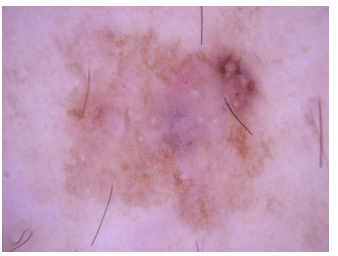
Training, validation, and test sets are available at the 2018 International Skin Imaging Collaboration (ISIC) challenge archive website [9] (see [2,15] for more information), which also includes a live challenge submission option, for continuous evaluation of automated classifiers using the dataset.



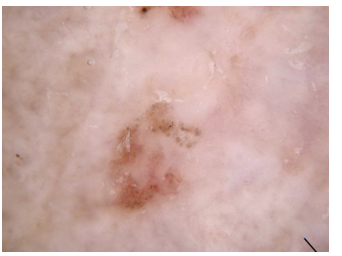
(a) Melanocytic nevi



(b) Melanoma

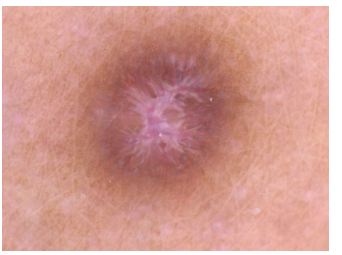


(c) Benign keratosis-like lesions

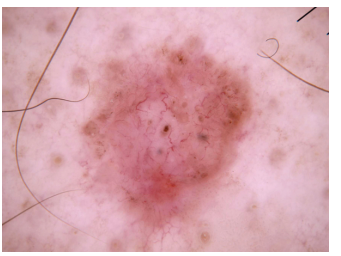


(d) Actinic keratoses

(e) Vascular lesions



(f) Dermatofibroma



(g) Basal cell carcinoma

Fig. 1: HAM 10000 dataset: examples of representative images for each class.

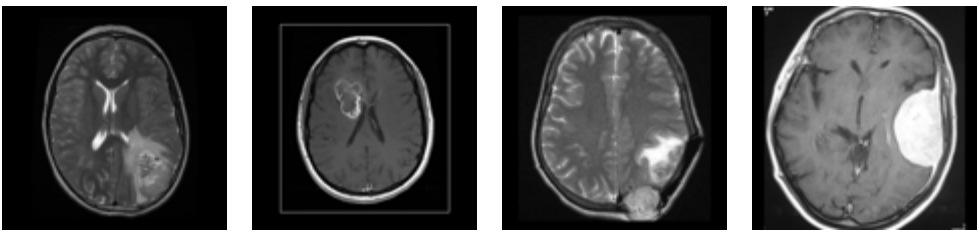


Fig. 2: Examples of tumorous brains.

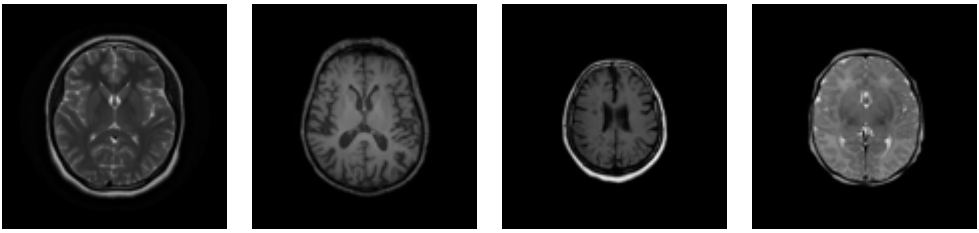


Fig. 3: Examples of healthy brains.

2.1.2. The Brain Tumor Dataset

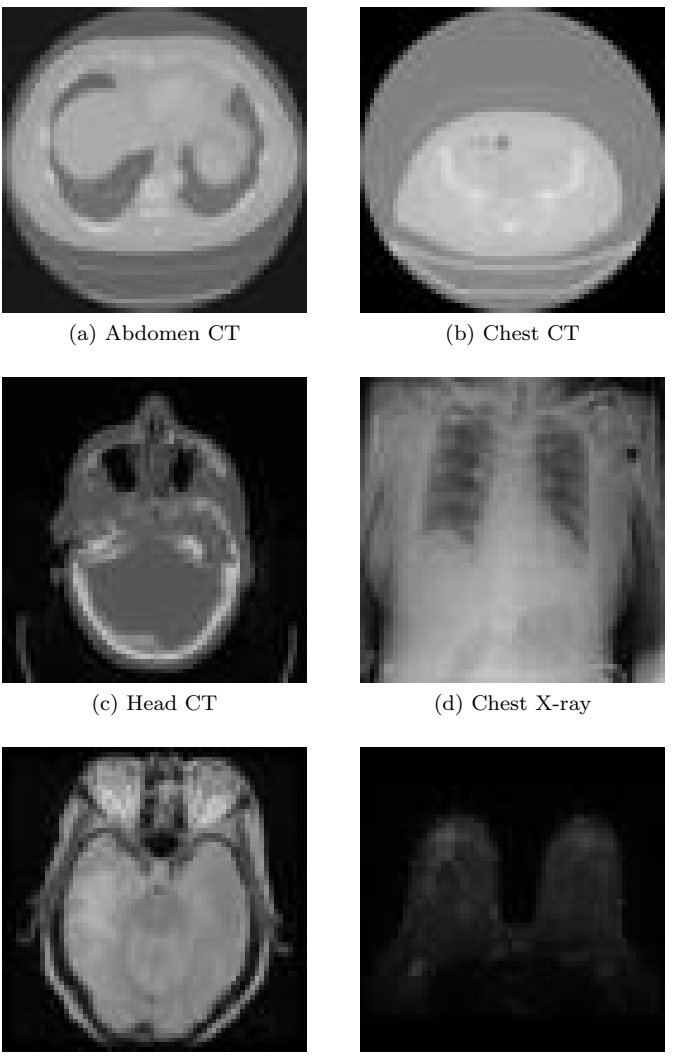
The Brain Tumor dataset [1] contains a total of 253 grayscale Brain MRI slices: 155 images that exhibit tumor and 98 slices that do not (Figures 2 and 3).

This dataset has been published on Kaggle [5] (arguably the world's most famous machine learning and data science community) and used in several experiments using different network architectures to test performance and accuracy in classification tasks.

2.1.3. The MedNIST Dataset

The MedNIST dataset [4] is a collection of grayscale medical images categorized into six different classes: *Abdomen computed tomography (CT)*, *Chest CT*, *Head CT*, *Chest X-ray*, *Brain magnetic resonance (MR)*, and *Breast MR* (Figure 4).

This dataset was created with the purpose of teaching basic deep learning concepts. In a follow up work [3], the dataset was used to demonstrate how to optimize a simple image classifier, thereby guiding researchers in the process of building an environment to execute a complete deep learning application, understanding the deep learning workflow, and focusing more specifically on parameters adjustments and their influence on the overall performance.



(a) Abdomen CT (b) Chest CT

(c) Head CT (d) Chest X-ray

(e) Brain MR (f) Breast MR

Fig. 4: MedNIST dataset: examples of representative images for each class.

2.2. CNN Architectures

CNNs have demonstrated excellent performance in image classification tasks. Classical examples of successful networks include AlexNet [10] and VggNet [14]. Modern CNN architectures are deeper and use more complex connections among layers, such as ResNet [7] and DenseNet [8].

When an input image is processed at the convolutional layer, the image goes through a series of convolution operations taking as input different pixels of the image. These operations are performed using masks (usually 3×3 or 5×5) in a similar way to spatial filters in classical image processing, except for two significant differences: (i) the masks' coefficients (weights) are learned by training the CNN (rather than fixed by the specific image processing technique); and (ii) the number of pixels (known as *stride*) by which a mask is shifted after performing the convolution in a certain portion of the image can be chosen by the CNN designer.

The choice of stride value in CNNs impacts the number of computations (additions and multiplications) required to process each image: smaller strides require more computation than larger strides. Typical stride values are 1, 2, and 3. A stride of 1 means that all pixels form the input image will be processed, which sets an upper bound on the computational cost of the convolution operations.

In this paper we extend the ideas first presented in [16] to explore and test the idea of varying the stride between 1 and 3 depending on the relative position of the pixels to be used in the convolution operation within the image: pixels that are closer to the center of the image will be processed using a smaller stride whereas pixels closer to the edges of the images are processed with a larger stride.

Figure 5 illustrates the process using a generic input image whose size is 15×15[1], where the blue pixels are pre-selected for the convolution operation. The resulting effect is the assignment of corresponding stride values as follows:

- pixels 1 to 5: stride 3,
- pixels 5 to 7: stride 2,
- pixels 7 to 9: stride 1,
- pixels 9 to 11: stride 2,
- pixels 11 to 15: stride 3.

Medical images sizes are usually larger, therefore the original strategy has to be expanded to accommodate larger images. It is easy to imagine how the process would scale up. The sampling process has to be done in a way where there is a given percentage of the pixels that would be sampled with stride 1, and others with strides 2 and 3. This has to be arranged in a way as to sample half of the pixels in each dimension. For

[1] In our experiments the same basic idea was adapted and implemented with different image sizes, depending on the dataset used.

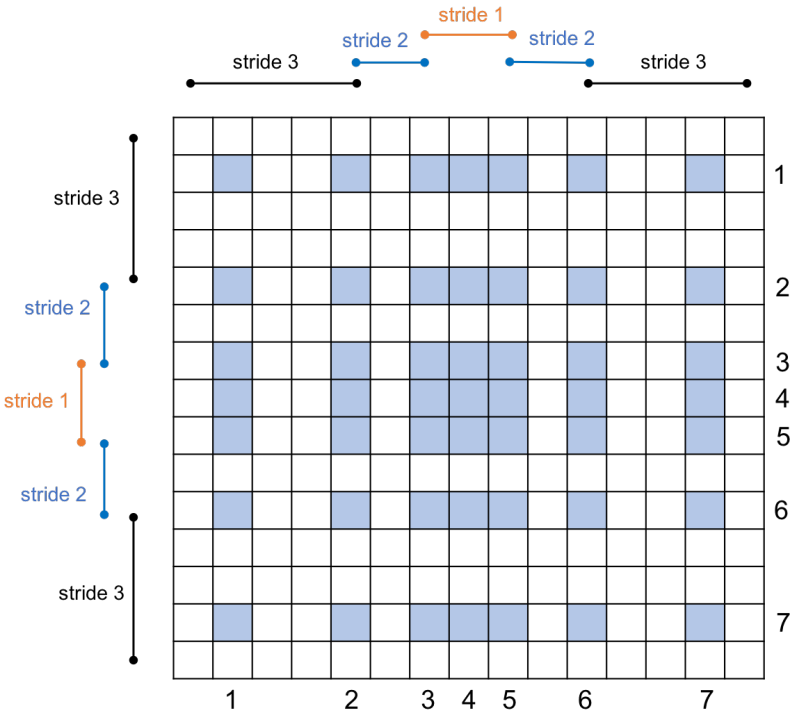


Fig. 5: Simple example of the proposed patterned stride mechanism for a generic image of 15×15 pixels.

example, in the MedNIST dataset all images are 64×64 pixels. The stride arrangement used for both dimensions in this dataset was as follows:

- pixels 1 to 8: stride 3,
- pixels 8 to 30: stride 2,
- pixels 30 to 35: stride 1,
- pixels 35 to 57: stride 2,
- pixels 57 to 64: stride 3.

In [16], the authors formalized their assumptions about the advantages of the patterned stride approach with two complementary testable hypotheses:

- H1: The use of patterned stride in images whose main contents are in the **central** portion of the image will lead to **improved** performance (when compared to the baseline case of comparable computational complexity, i.e., fixed stride = 2).

- H2: The use of patterned stride in images whose main contents are **not** in the **central** portion of the image will lead to **decreased** performance (when compared to the baseline case of comparable computational complexity, i.e., fixed stride = 2).

In this paper we focus on new examples of successful use of patterned stride, introduce and explain three new hyperparameters (P_1, P_2, P_3), and present image score calculations to provide a rough estimate of how effective the patterned stride mechanism can be in different datasets.

2.3. Hyperparameters P_1, P_2, P_3

One of the assumptions that are needed to compare this new method against fixed stride 2 is to create a model which has similar computational and memory requirements. In order for this to happen, the first layer has to reduce the sampling points by half. The sampling points can be spread in different ways throughout the image. The proposed method postulates a concentrated sampling in the center and sparse at the edges, varying the stride from 1 to 3 in different parts of the images as seen in figure 5. However, the number of pixels in each area (stride 1, 2 or 3) can vary while still keeping the same total number of sampling points. Figure 6 shows an example where the stride 2 area is enlarged compared to strides 1 and 3. We propose the introduction of three new hyperparameters, that will define the percentage of sampling points in each stride region, P_1, P_2 and P_3 which are the percentage of total sampling points in each corresponding stride region (1, 2 and 3) respectively.

Since we still need to comply with the requirement of having half of the sampling points in the first convectional layer, we have to guarantee that the number of sampling points on stride regions 1 and 3 are the same. Another intuitive way to explain this is that any increase in a more concentrated sampling area (stride 1) has to be compensated by an increase of a sparse area (stride 3).

Therefore the choice of P_1, P_2, and P_3 is bound by the following conditions:

$$P_1 = P_3 \ ,$$

and

$$P_1 + P_2 + P_3 = 1 \ .$$

The optimal values used for P_1, P_2, and P_3 for each experiment will be described in Section 3.

2.4. Image Score

To guide the selection of hyperparameters P_1, P_2, and P_3, we created a new method to calculate the input images' score. The objective of the score is to calculate how the meaningful part of the image frame is concentrated in the center. Image scores

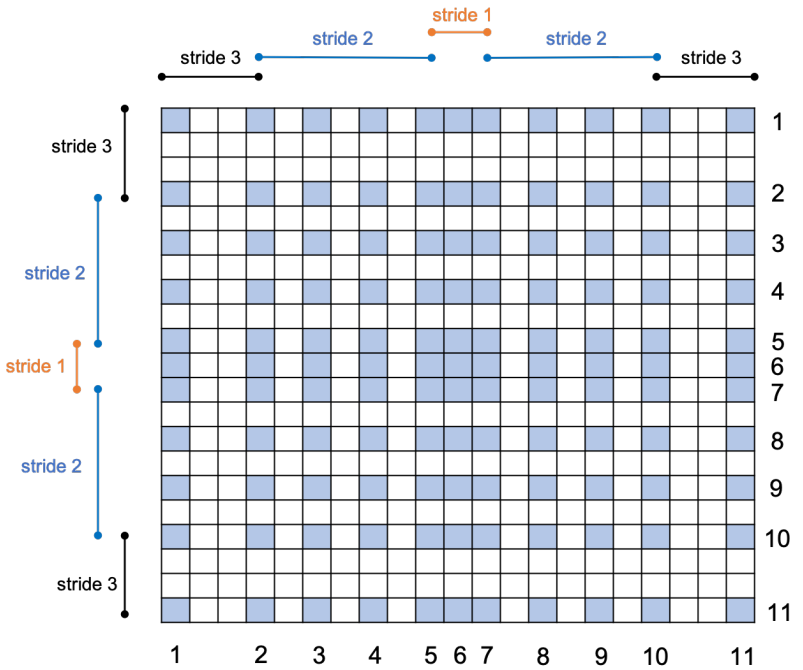


Fig. 6: Simple example of variation of stride areas for a generic image of 21×21 pixels.

range between 0 and 1, where 0 is an image where all the information is completely concentrated in the center and 1 means the opposite. Figure 7 shows an example of image score calculation using a quadrilateral bounding polygon, where the meaningful part of the image is shown in yellow.

The score is calculated as:

$$S = \frac{1}{2}\left(\frac{x_c + y_c}{2} + \sum_{k=1}^{s} \frac{x_k + y_k}{2n}\right) ,$$

where n is the number of sides of the bounding polygon and x_c, y_c, x_k, and y_k are the partial scores of each individual point calculated as follows:

$$x_c = \frac{|p_{cx} - i_{cx}|}{d_x} ,$$

$$y_c = \frac{|p_{cy} - i_{cy}|}{d_y} ,$$

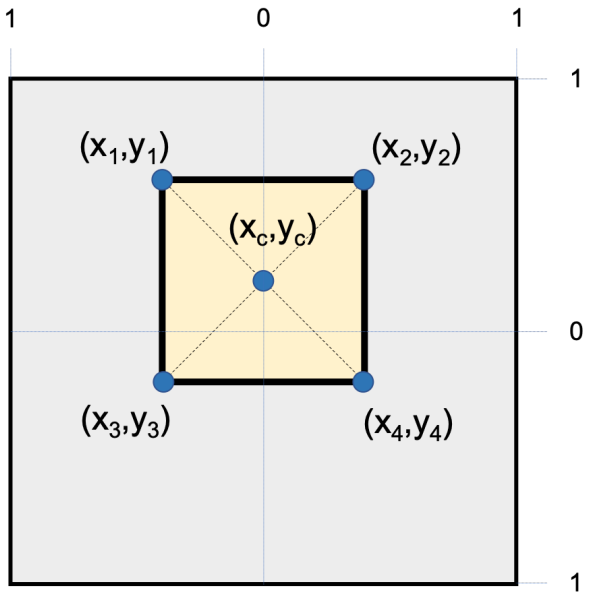


Fig. 7: Image score bounding box.

$$x_k = \frac{|p_{kx} - i_{cx}|}{d_x} \ ,$$

$$y_k = \frac{|p_{ky} - i_{cy}|}{d_y} \ ,$$

where

p_{cx} – x coordinate of bounding polygon center,
p_{cy} – y coordinate of bounding polygon center,
p_{kx} – x coordinate of k^{th} point of the bounding polygon,
p_{ky} – y coordinate of k^{th} point of the bounding polygon,
i_{cx} – x coordinate of frame center,
i_{cy} – y coordinate of frame center,
d_X – image width in pixels,
d_Y – image height in pixels.

This method was used to calculate the scores for the datasets used in the experiments and will be reported together with the results of each experiment in Section 3.

3. Experiments and Results

This section describes in detail the experiments performed to test the performance of the patterned stride mechanism in a simple CNN (Figure 8), whose layers are described below:

1. Convolutional 2D layer: this is the layer in which we modify the stride parameter,
2. Batch Normalization layer,
3. ReLu layer,
4. Max Pooling layer,
5. Convolutional 2D layer,
6. Batch Normalization layer,
7. ReLu layer,
8. Max Pooling layer,
9. Convolutional 2D layer,
10. Batch Normalization layer,
11. ReLu layer,
12. Fully connected layer,
13. Softmax layer,
14. Classification layer[2].

3.1. Experimental setups

All experiments were performed using MATLAB. Since MATLAB doesn't provide an option to select variable strides for convolutional layers, some modifications in the MATLAB source code were needed in other to perform the desired operation.

The train/test split was 95/5, i.e., 95% of the images were used for training/validation and 5% for tests. Categorical cross-entropy loss and stochastic gradient descent with momentum (SGDM) optimizer were used.

Network parameter values were experimentally selected to achieve the best performance for each task. For the HAM10000 dataset, the network was trained using 20 epochs and a decaying learning rate (LR), starting with 0.0003 and reducing it by half every 2 epochs. For the Brain Tumor dataset, we used 40 epochs and a LR of 0.0003. For the MedNIST dataset, we used 8 epochs and a fixed LR of 0.0003.

3.2. Results

Test runs were performed in four different network configurations, where the stride parameter was set to 1, 2, 3 or *patterned*, depending on the run. Final results for the

[2]The number of nodes in this layer will vary depending on the experiment according to the number of classes.

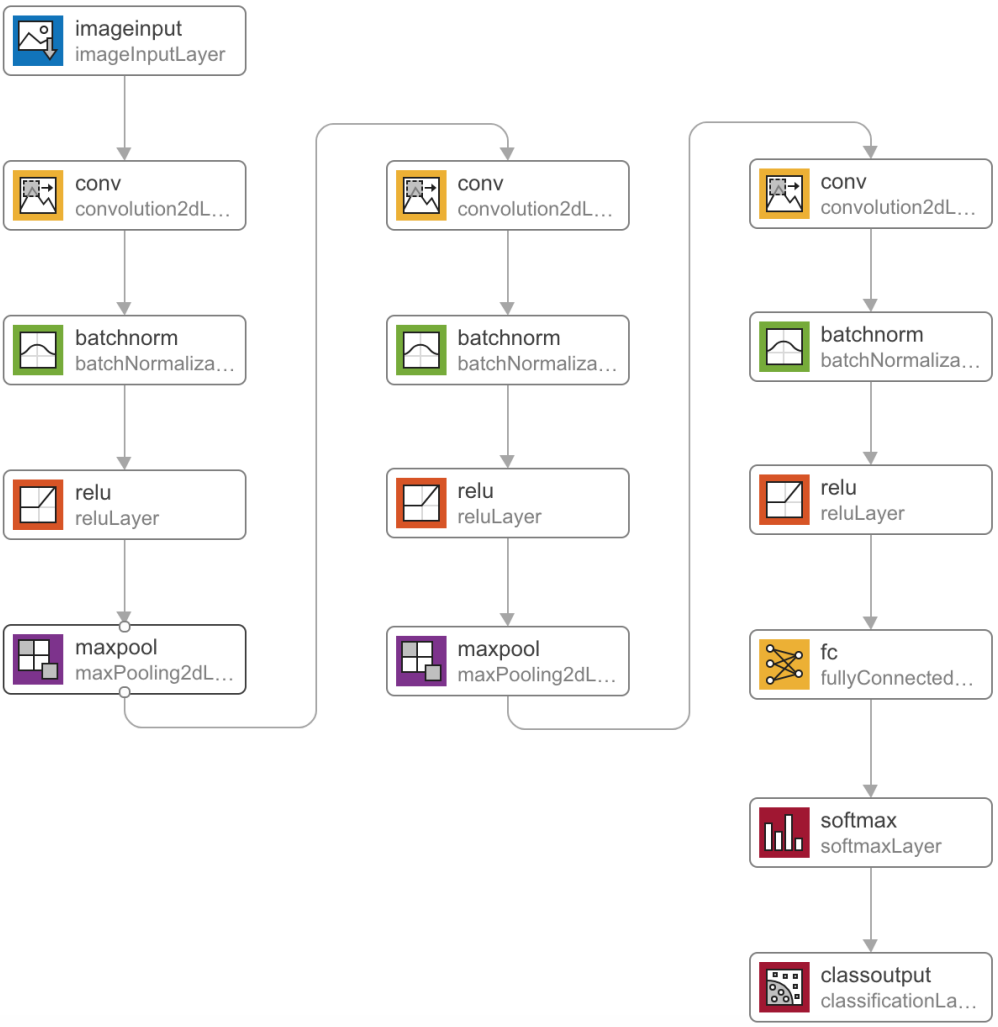


Fig. 8: CNN architecture used for the classification experiments.

Tab. 1: Skin lesion classification using the HAM10000 dataset and patterned stride: classification accuracy results and time measurements.

Stride	Accuracy	Training Time	Inference Time
1	77.0 %	258 min	31.9 ms
2	76.4 %	105 min	10.9 ms
3	75.5 %	70 min	5.8 ms
Patterned	78.1 %	105 min	10.9 ms

Tab. 2: Brain tumor detection using the Brain Tumor dataset and patterned stride: classification accuracy results and time measurements.

Stride	Accuracy	Training Time	Inference Time
1	89.6 %	12.5 min	50.6 ms
2	84.0 %	4.2 min	15.3 ms
3	81.6 %	2.4 min	7.4 ms
Patterned	87.2 %	4.2 min	15.3 ms

HAM10000 dataset, Brain tumor dataset, and MedNIST dataset are shown in Tables 1, 2, and 3, respectively. In each table the respective classifier's accuracy and the elapsed time for training and inference for each case are reported. The tables confirm the computational cost for patterned stride is comparable to the cost for stride 2.

We performed tests with different values of hyperparameters P_1, P_2, and P_3 and reported results for optimal parameter selection. The optimal values for each dataset are shown in Table 4.

Tab. 3: Image modality classification using the MedNIST dataset and patterned stride: classification accuracy results and time measurements.

Stride	Accuracy	Training Time	Inference Time
1	99.8 %	16.5 min	18.6 ms
2	98.3 %	5.0 min	6.1 ms
3	95.9 %	2.4 min	3.5 ms
Patterned	98.9 %	5.0 min	6.1 ms

Tab. 4: Optimal values for $P1$, P_2, and P_3 of each dataset.

Dataset	P_1	P_2	P_3
HAM10000	0.24	0.52	0.24
Brain Tumor	0.46	0.08	0.46
MedNIST	0.16	0.68	0.16

4. Discussion

Experimental results on three different classification tasks, in three different datasets, have confirmed the hypothesis that the proposed patterned stride mechanism outperforms the fixed stride options (with stride equal to 2 or 3) in all test cases.

For the HAM10000 dataset, the results were even better, with an accuracy even higher than the stride 1 configuration (at a fraction of the computational cost). This is consistent with the fact that the most informative portion of the skin lesion images in the HAM10000 dataset is usually centered (and the surrounding area contains very little information), which is the optimal case for the proposed patterned stride scheme.

On a related note, figures 9 and 10 show the confusion matrices for the case of patterned stride and stride 2 in the MedNIST classification task. In Figure 9, which displays the results for patterned stride, one can see that two of the images from the brain MR category were misclassified, one as head CT and the other as breast MR. In Figure 10, which displays the confusion matrix for stride 2, one can observe that there were two instances of misclassification between the CTHead and MRBrain classes, which did not happen for the patterned stride case (Figure 9). This result is particularly interesting because for these two classes, since the general brain shape is the same, the differences had to be found in internal image elements. This confirms that the patterned stride has a better performance when the central area of the image contains the most informative pixels within the image.

Even though the patterned stride mechanism delivered a better classification accuracy in the MedNIST classification task, there were some cases where it classified an image incorrectly. One example is shown in figure 11, where an image from the brain magnetic resonance class was classified as breast magnetic resonance. In this case, the input image was very dark and the key to classify it correctly lies on the observation of the external shape where the breast magnetic resonance has some distinguishing factors. Since the patterned stride network concentrates its effort in central places, it wasn't able to get the necessary nuances of the image for a correct classification.

Another interesting observation is the relationship between image scores and hyperparameters P_1, P_2, P_3. Table 5 shows the comparison of image scores for each dataset with the optimal values used for the parameters and indicates that larger scores need

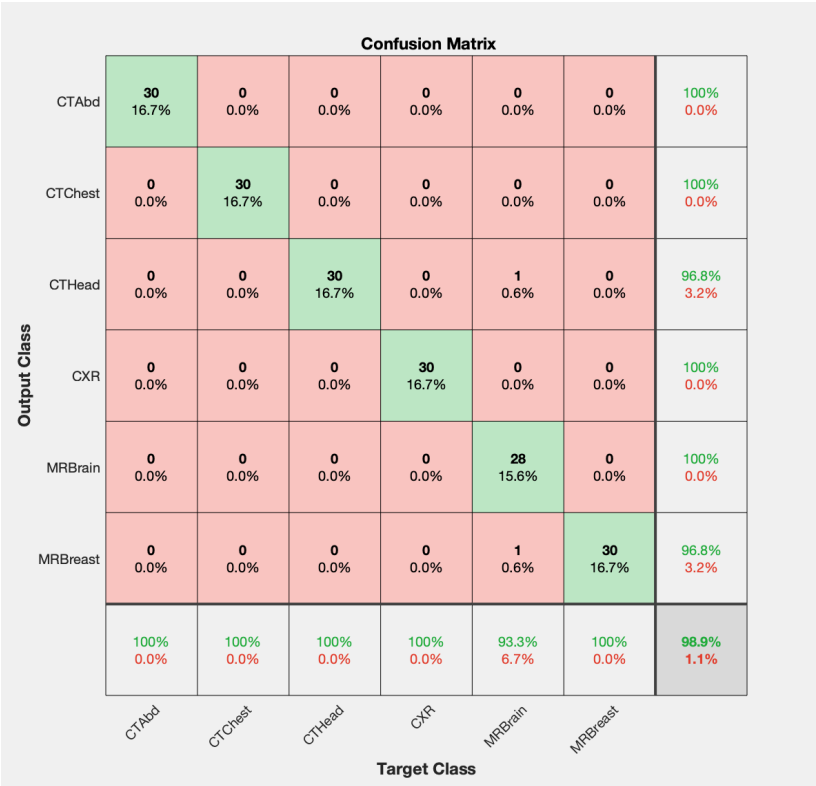


Fig. 9: Confusion matrix for the patterned stride case using the MedNIST dataset.

Tab. 5: Comparison between image scores and hyperparameters P_1, P_2, and P_3 for each dataset.

Dataset	Image Score	P_1	P_2	P_3
Brain Tumor	0.29	0.46	0.08	0.46
HAM10000	0.35	0.24	0.52	0.24
MedNIST	0.44	0.16	0.68	0.16

to use smaller values for P_1 and P_3 and larger values for P_2. These results confirm the usefulness of the image scores as a preliminary estimate of the effectiveness of the patterned stride mechanism across different datasets.

The proposed method has its limitations, more notably: (i) it does not provide better

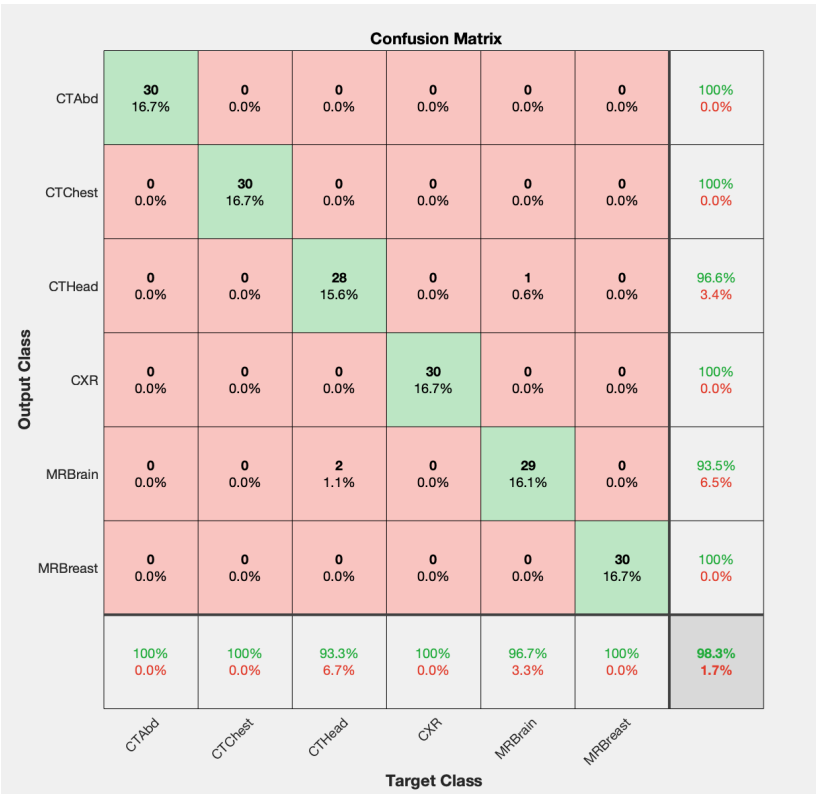


Fig. 10: Confusion matrix for the stride 2 case using the MedNIST dataset.

results than the fixed stride 2 baseline in cases where the informational content of an image is spread throughout the image (as opposed to concentrated on its center)[3]; and (ii) it does not work for cases where most of the useful information is concentrated in a small portion of the image, but away from its center.

To address (i), a simple two-class classifier could be used to determine if a dataset (and associated classification problem) is suitable for the patterned stride mechanism or not. Such classifier could use an easy-to-compute measure of image complexity or homogeneity (e.g., entropy) as its main feature. In its simplest form, if the entropy is higher than a certain threshold, the dataset is not a good candidate for the patterned stride approach. More sophisticated features and classifiers, of course, could be used.

A potential solution to (ii) could be the use of alternative sampling patterns (see

[3]See counterexamples in [16] that support hypothesis H2.

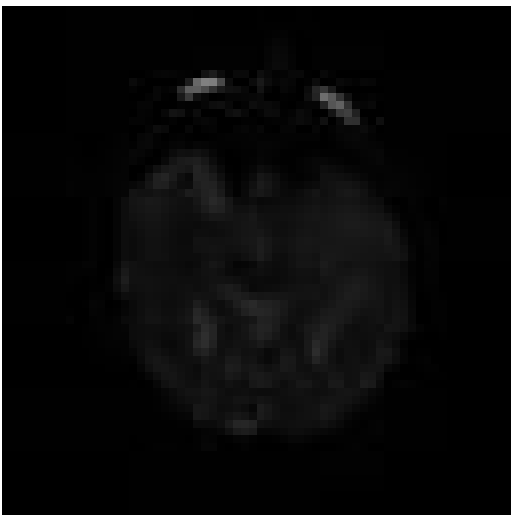


Fig. 11: Brain magnetic resonance classified as breast magnetic resonance by the patterned stride network.

Figure 12 for examples of five different predefined patterns[4] for *centered* as well as *top-right*, *top-left*, *bottom-right*, and *bottom-left* cases) and a selector algorithm that could be used to determine – for each individual image in the dataset – which of the (five) patterns is most appropriate. Such algorithm could use any type of region-of-interest (ROI) detection scheme, e.g., off-the-shelf face detectors for images involving a face, or ROI computation using saliency maps [13] for cases where the object of interest is also the most salient in the scene.

5. Conclusion

We have extended a method for implementing a patterned stride mechanism in CNNs and successfully demonstrated experimentally that the use of patterned stride leads to higher accuracy than a fixed stride baseline case of same computational complexity in three different medical image classification tasks and datasets.

The proposed approach could be extended to other datasets, more complex CNN architectures, and different tasks using CNNs, such as semantic segmentation and object detection.

[4]The examples show the sampling distribution for a 15×15 image, but can be extrapolated to any image size.

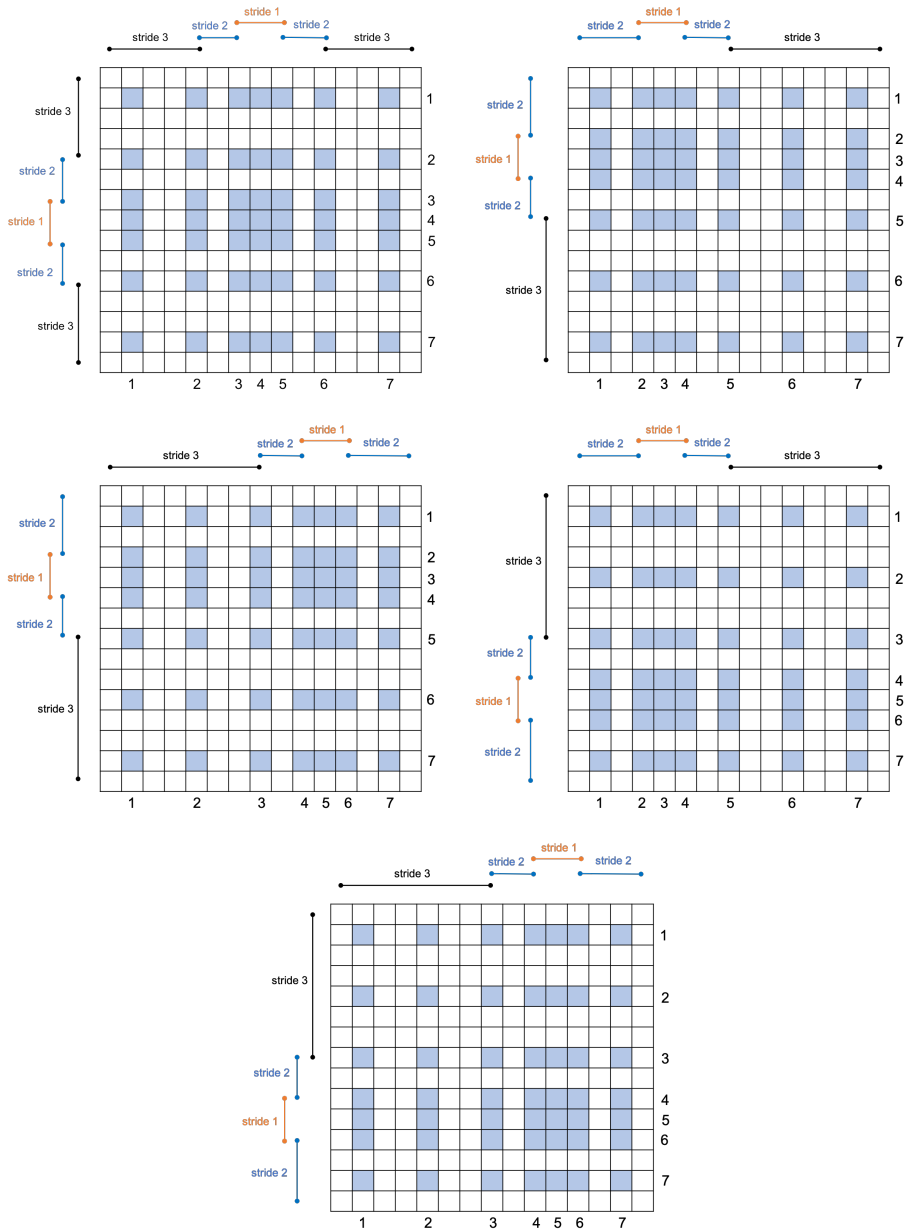


Fig. 12: Adjustable patterned stride sampling patterns.

References

[1] N. Chakrabarty. Brain MRI images for brain tumor detection, 2019. `https://www.kaggle.com/navoneel/brain-mri-images-for-brain-tumor-detection`. Dataset [accessed Feb 2021].

[2] N. Codella, V. Rotemberg, P. Tschandl, et al. Skin Lesion Analysis Toward Melanoma Detection 2018: A Challenge Hosted by the International Skin Imaging Collaboration (ISIC). *arXiv preprint*, 2019. arXiv:1902.03368.

[3] B. J. Erickson. Magician's Corner: 2. Optimizing a simple image classifier. *Radiology: Artificial Intelligence*, 1(5):e190113, 2019. doi:10.1148/ryai.2019190113.

[4] B. J. Erickson. Magician's Corner: How to start learning about deep learning. *Radiology: Artificial Intelligence*, 1(4):e190072, 2019. doi:10.1148/ryai.2019190072.

[5] A. Goldbloom, B. Hamner, J. Moser, et al. Kaggle: Your Machine Learning and Data Science Community. `https://www.kaggle.com`. [accessed Feb 2021].

[6] C. Guo, Y.-l. Liu, and X. Jiao. Study on the influence of variable stride scale change on image recognition in CNN. *Multimedia Tools and Applications*, 78(21):30027–30037, 2018. doi:10.1007/s11042-018-6861-0.

[7] K. He, X. Zhang, S. Ren, et al. Deep residual learning for image recognition. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR 2016*, pages 770–778, Las Vegas, NV, USA, 27-30 Jun 2016. doi:10.1109/CVPR.2016.90.

[8] G. Huang, Z. Liu, L. Van Der Maaten, et al. Densely connected convolutional networks. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR 2017*, pages 4700–4708, Honolulu, HI, USA, 21-26 Jul 2017. IEEE. doi:10.1109/CVPR.2017.243.

[9] H. Kittler, N. C. F. Codella, M. E. Celebi, et al. ISIC 2018: Skin Lesion Analysis Towards Melanoma Detection. `https://challenge2018.isic-archive.com`.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in neural information processing systems*, volume 25, pages 1097–1105. Curran Associates, Inc., 2012. `https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf`.

[11] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015. doi:10.1038/nature14539.

[12] G. Litjens, T. Kooi, B. E. Bejnordi, et al. A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42:60–88, 2017. doi:10.1016/j.media.2017.07.005.

[13] O. Marques, L. M. Mayron, G. B. Borba, and H. R. Gamba. Using visual attention to extract regions of interest in the context of image retrieval. In R. Menezes, editor, *Proc. 44st Ann. Southeast Regional Conf.*, pages 638–643, Melbourne, FL, USA, 10-12 Mar 2006. ACM. doi:10.1145/1185448.1185588.

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

[15] P. Tschandl, C. Rosendahl, and H. Kittler. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific Data*, 5:180161, 2018. doi:10.1038/sdata.2018.161.

[16] L. Zaniolo and O. Marques. On the use of variable stride in convolutional neural networks. *Multimedia Tools and Applications*, 79:13581–13598, 2020. doi:10.1007/s11042-019-08385-4.

Luiz Zaniolo received his Master's degree in Computer Engineering from Florida Atlantic University, Boca Raton, FL, USA and is pursuing a Ph.D. degree in Computer Science at the same institution.

Oge Marques received his Ph.D. degree in 2001 from Florida Atlantic University (FAU), Boca Raton, FL, USA. He is Professor of Computer Science and Engineering at FAU since 2001. He is the author of 11 books and more than 120 scholarly publications in the area of Visual Artificial Intelligence. Dr. Marques is a Sigma Xi Distinguished Speaker, a Fellow of the Leshner Leadership Institute of the American Association for the Advancement of Science (AAAS), Tau Beta Pi Eminent Engineer, and a Senior Member of both the IEEE (Institute of Electrical and Electronics Engineers) and the ACM (Association for Computing Machinery).

Chemical Ripening and Contaminations Detection Using Neural Networks-based Image Features and Spectrometric Signatures

R. Roopalakshmi
Manipal Institute of Technology, Manipal, India.
roopalakshmi.r@manipal.edu

Abstract. In this pandemic-prone era, health is of utmost concern for everyone and hence eating good quality fruits is very much essential for sound health. Unfortunately, nowadays it is quite very difficult to obtain naturally ripened fruits, due to existence of chemically ripened fruits being ripened using hazardous chemicals such as calcium carbide. However, most of the state-of-the art techniques are primarily focusing on identification of chemically ripened fruits with the help of computer vision-based approaches, which are less effective towards quantification of chemical contaminations present in the sample fruits. To solve these issues, a new framework for chemical ripening and contamination detection is presented, which employs both visual and IR spectrometric signatures in two different stages. The experiments conducted on both the GUI tool as well as hardware-based setups, clearly demonstrate the efficiency of the proposed framework in terms of detection confidence levels followed by the percentage of presence of chemicals in the sample fruit.

Key words: chemical ripening, arsenic contamination, visual features, IR spectral signatures.

1. Introduction

Nowadays health is of important concern for everyone, hence eating good quality fruits is a primary requirement for sound health. The fruits in general are plant products containing sugar, vitamin C and water along with minerals, cellulose, protein and photo chemicals that protect human body against various diseases [6]. In general, fruits obtain desirable flavor, quality, color and other textural changes during their natural ripening process. Unfortunately, nowadays it is quite very difficult to obtain naturally ripened fruits, due to existence of huge numbers of chemically ripened fruits in the markets, which are being ripened using hazardous chemicals such as calcium carbide (CaC_2). For example, nearly 80% fruits such as mango, papaya and banana are artificially ripened using different chemicals [16].

In general, though fruit ripening is a natural process, in order to speed up the rate of fruit ripening, most of the farmers and vendors use artificial ripening agents like calcium carbide. Specifically, calcium carbide is a dangerous, corrosive chemical and regular consumption of it leads to vomiting, diarrhoea, eye damage, ulcers, hypoxia and neurological disorders, and even to cancer due to the presence of arsenic as well as phosphorous poisoning traces. Due to these reasons, as per PFA (Prevention of Food

Adulteration) act in 1955, chemical ripening of fruits is strictly banned. Though calcium carbide is banned, still some traders are employing chemical ripening for profit purposes.

Spacial care must be paid to climacteric fruits. Climacteric fruits are those which, beginning from a certain developmental stage, continue to develop to full maturity, even when harvested. Specifically, in India, most of climacteric fruits, such as mango, banana and papaya are chemically ripened with industrial grade calcium carbide [16]. Specifically, in India, calcium carbide, a carcinogen, is widely used for artificial ripening of fruits such as banana as well as mango, which is illegal and strictly banned. Therefore, the identification of artificially ripened fruits followed by the quantification of CaC_2 contamination in such fruits is very important in order to safeguard the consumers from a series of health problems. Based on these aspects, a new framework is introduced in this paper, which makes use of both computer vision as well as Near-Infrared (NIR) spectrometric techniques for detecting the artificially ripened fruits, followed by the computation of chemical contaminations present in the sample fruits.

2. Related Work

In the existing literature, computer vision-based techniques are popularly utilized for quality determination and grading of fruits by means of automating the grading processes as well as minimizing the monotonous inspection tasks. Further, computer vision is also widely employed in the literature for defect detection, and classification of ripeness of fruits based on their appearance. For example, a comparative study of vitamins A, B, and C content in different types of tomatoes including ethylene and vine-ripened tomatoes is presented in [6]. Ahmad et al. [1] analysed the effect of ethylene towards the speed of ripening as well as the quality of banana fruit; however, they failed to accurately discriminate between ethylene vs. non-ethylene treated bananas. In [4], the authors employed acoustic responses, nuclear magnetic resonance and optical properties in order to estimate the firmness of fruits; yet, the presented method failed to predict the chemical ripening of fruits. In 2015, Bhosale et al. [3] presented a capacitive sensing system using color indexing and echo measurements, which can detect different ripening stages of papaya fruit. Recently, Pratim Ray et al. [15] introduced a monitoring tool for finding the ripening stage of banana fruit using color indices, which can also send the ripening information to the monitoring person present in a remote area with the help of a GSM module.

In [7], the authors introduced threshold-based segmentation method using Haar features, to detect chemical ripening of banana fruits. Though the authors employed the third level of decompositions in wavelet domain for analysis of discriminatory behaviors, the proposed method suffered due to the variations in the features of ripened bananas. Thermal imaging framework for chemical ripening was proposed by Ansari in [2], which

utilized infrared energy emitted by the sample fruit for pre-processing and segmentation followed by feature extraction stages. However, the performance of this method is slightly lower when compared to other methods due to the complex nature of neural network strategies used in the system. Further, Sukhesh et al. [9] introduced a cost effective device using sensors, which was capable of detecting nutrients and chemical contents in vegetables and fruits and of presenting it in the display on smart phones. Salunkhel and Aniket [17] presented a computer vision-based system, which could detect various ripening stages of Mango fruit using RGB and HSV features of images. The proposed system classified only the ripening stages of mango.

Veena and Bhat [5] designed a simple portable instrument for the detection of chemically ripened banana fruits using color-based features, which can also find out the specific ripening stage of the sample banana fruit. Therefore, this method performs better in terms of detecting chemical ripening, yet it suffers in case of complex banana structures. In [8], an IR-based sensor system was introduced, which can detect the presence of ethylene, so that different fruit ripening stages are clearly classified. Specifically, the authors used the thermal emission concept for estimating ethylene release during fruit ripening process. Although this method provides good reproducibility, yet it concentrates primarily on discrimination of fruit ripening stages to ensure food safety.

In the existing literature, only few efforts are made towards identifying the chemically ripened fruits by employing hybrid techniques including computer vision and sensor techniques [14]. For instance, Verma and Hegadi [20] presented a remote monitoring system for banana ripening process by employing wireless networks, which helps the user to monitor the ripening from a remote place. Recently, Srividya et al. [18] proposed an ethylene measurement system in order to predict correct stage of ripening of fruits using image-based features. In [13], the authors developed a mobile-based interface for detecting chemically ripened fruits, which performs histogram comparisons in order to obtain detection results. Although this method is easy to use, it still performs slightly low, due to the usage of mere surface features of the sample fruits. Recently, in [11] and [10], the authors employed NIR spectroscopic method as well as gold nano particle-based techniques in order to detect chemically ripened mango fruits. However, these techniques fail to quantify the presence of arsenic in the chemically contaminated fruit, which is yet to be explored in detail in the existing literature.

To summarize, most of the state-of-the art techniques are primarily focusing on detecting the stages of fruit ripening, as well as on the identification of chemically ripened fruits with the help of computer vision-based methods. From another perspective, image features are less effective in quantification of the exact amount of chemical contaminations present in the sample fruits. Due to these issues, *promising frameworks are very much essential, which employ both the image-based features and IR spectrometric signatures in order to detect artificial ripening of fruits followed by computation of chemical contaminations present in the sample fruits.*

3. Motivation and contributions

In this paper a new framework is introduced for detecting the man-made ripened fruits followed by the computation of chemical contaminations present in the fruit by employing both the spectrometric features as well as visual feature descriptors. Specifically, the new framework named ***Chemical Ripening and Contamination Detection (CRCD)*** is introduced, which first identifies the artificially ripened fruits and then quantifies the presence of chemical contaminations (in terms of presence of arsenic) in the given sample fruit. More specifically, the primary contributions of the proposed framework are as follows.

- A new *GUI-based Artificial ripening detector tool* for banana fruit is developed which is used to identify unnaturally ripened banana fruit by making use of edge and histogram-based visual feature descriptors. Further, the prototype of this GUI tool is also evaluated in a web-based portal and mobile-based interfaces in order to facilitate remote access, which is illustrated in detail in Section 7.
- A novel *arsenic contamination detection setup* is introduced which makes use of IR signature spectra of fruits for detecting chemically ripened fruits followed by Green Fluorescent Protein-based turbidity measurements for accurately quantifying the arsenic content present in the sample fruit, which is detailed in Sections 6.1 and 7.4.
- Further, chemical contamination rate of the given sample fruit is clearly indicated in terms of percentages in the *specially designed display panel fitted with aqua fruit chamber*, which in future can be employed effectively to protect customers from hazardous health issues. The setup of the aqua chamber and the panel is described in Sections 6.2, 6.3 and the detection results are described in Sections 7.3 and 7.4.

4. Methodology of the proposed framework

The block diagram for the proposed Chemical Ripening and Contamination Detection (CRCD) framework, is indicated in Fig. 1. It is implemented in two different modules, namely the image processing module and the arsenic detection module, which are detailed as follows.

In the first module of the proposed CRCD framework the captured images of sample fruit undergo the first stages of processing. The Near Infra-Red (NIR) camera is employed to capture the images of the sample fruits in different directions and views, including front and top views. Specifically, in the first stage, convolutional neural network (CNN) based classification algorithms are used to train both ripened and unripened categories of input banana images. More specifically, visual feature descriptors of input images including shape, edge and color features (weighted at the ratio 25:25:50) are extracted and classified by employing Inception v3 algorithm [19], which is one of the widely-used image recognition models. Initially, the pre-trained neural network extracts

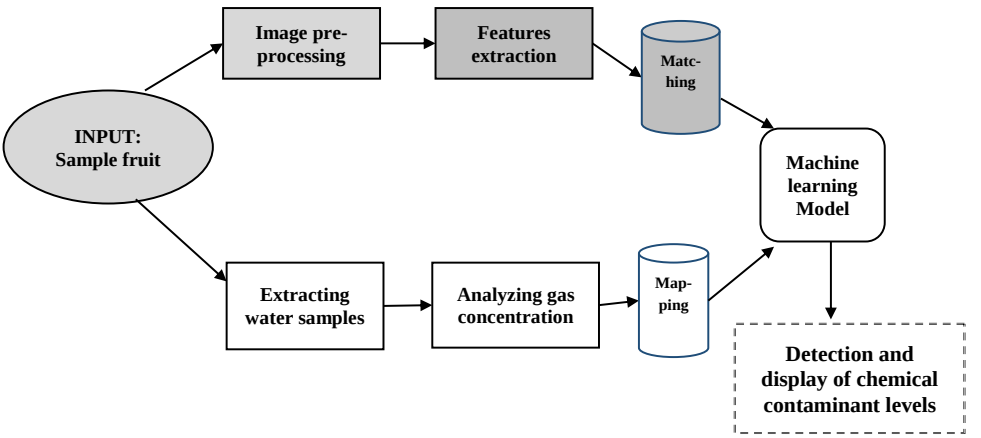


Fig. 1. Block diagram of the proposed CRCD framework.

the visual features using CNN including fully-connected and softmax layers. Then, the resultant features are combined to form a feature database named the Visual Features Database in order to proceed with testing stage.

In the testing stage, initially visual features of query banana image are compared with the corresponding features of the training database. The results of the comparison are represented in terms of confidence level metrics. In the proposed framework, 75% of the samples are utilized for the training stage after random selection whereas the remaining 25% of the samples are included in the testing part of the database.

In the second module, Green Fluorescent Protein (GFP) [12] based arsenic gas setup incubated with a water sample is utilized, in which the fluorescence is detected optically and quantified, so that the concentration of arsenic in the water samples can be measured. Precisely, the intensity of the fluorescence is a function of the amount of bacteria present in the water sample, which is further quantified to measure the turbidity of the water sample. The turbidity measurement enables us to determine chemical contaminations of the sample fruit in terms of arsenic gas contaminations. The resultant turbidity values are mapped and analyzed with standard metric values of normal water, in order to identify whether the fruit is a naturally or chemically ripened one. The comparison results are combined using the machine learning model and displayed in the LCD display panel mounted at the front display panel in the setup framework.

These processes are discussed in detail in the further Sections.

Fig. 2. (**a**) Sample snapshots of naturally ripened bananas and (**b**) sample snapshots of artificially ripened bananas.

5. Data bases

5.1. Banana fruits database creation

For experimental purpose, 500 banana images from 10 bananas of type Elaichi (species name: *Musa acuminata*) are captured under different scenarios including ripened, unripened, individual and group basis of fruits. Specifically, the banana dataset includes snapshot of bananas, which are captured using Canon 700D DSLR camera with the resolution of 4898×3265 pixels. Initially, few samples of unripened bananas are treated for artificial ripening with the help of calcium carbide. Precisely, the sample bananas are kept in an air tight container inside a dark room with the presence of calcium carbide for 8-10 hours, in order to make them ripen at a faster rate. The rest of the bananas are allowed to undergo natural ripening stages for a waiting period of 24 to 30 hours. Fig. 2a represents the snapshots of bananas, which were allowed to complete their ripening stages at normal conditions, whereas Fig. 2b indicates the sample snapshots of bananas, which were treated with calcium carbide for completing their ripening stage, respectively.

5.2. Mango fruits database creation

To evaluate the performance of the proposed framework towards mango fruits, a mango fruit dataset consisting of images was generated for both training and testing purposes. Precisely, 60 mangoes belonging to four different varieties: Alphonso, Badam, Mallika and Neelam are considered. One set of mango fruits are allowed to ripen naturally while the other set of mangoes were artificially ripened using artificial ripening agents like calcium carbide. Fig. 3 shows sample snapshots of mango images in various views.

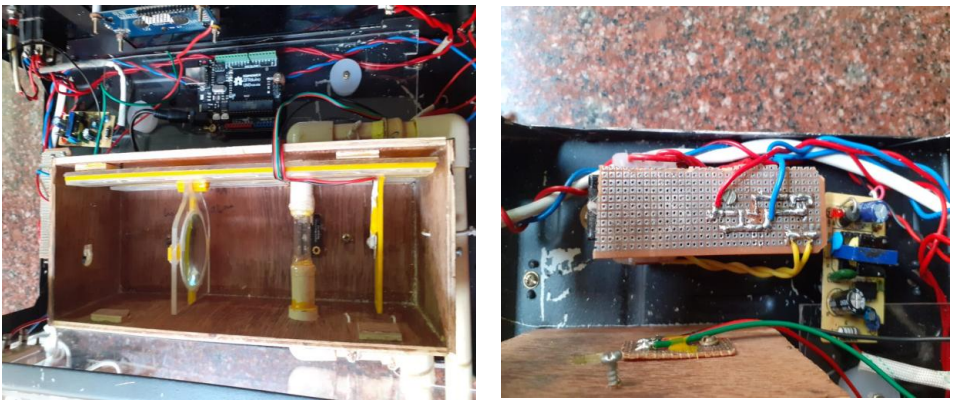
Fig. 3. Sample snapshots of mango images in various views.

Further, nearly 950 fruit images were generated for both training and testing datasets, which were labeled separately as naturally ripened mango fruits as well as artificially ripened fruits.

6. Experimental setup

6.1. Arsenic contamination detection setup

The basic principle of this setup is based on a fluorescent method, where GMO bacteria-based Green Fluorescent Protein-based bacterial biosensor (GFP) [12] is used to detect the presence of arsenic in the test water sample. Then the optical detection of fluorescence used to quantify the result in terms of the concentration of arsenic. Specifically, a vial containing the test water sample is placed on a socket and a fluorescent excitation LED light is passed through it, which also includes $\lambda = 488\,\text{nm}$ needed for enhanced GFP (eGFP). More specifically, the GFP absorbs blue light ($\lambda = 475\,\text{nm}$) and emits green light ($\lambda = 504\,\text{nm}$) which is detected by a photosensor in order to deal with the intensity loss. Then the eGFP fluorescence signal reaches the photosensor with the help of a long-pass filter. Precisely, the intensity of the fluorescence can be indicated as a function of the number of bacteria present in the sample. More precisely, the number of bacteria can be quantified indirectly by measuring the turbidity of the sample [12]. A red LED is placed in-line with the photosensor so that the transmittance can be measured and converted into turbidity. The measurement of turbidity makes it possible to normalize the results with respect to the density of bacteria. Specifically, the measurement of turbidity can be employed in order to determine the concentration of arsenic present in the water sample and thereby chemical contaminations in the fruit can be detected. More specifically, the transmission results are compared against a standard curve of water containing known



(a) **(b)**

Fig. 4. (**a**) Snapshot of the GPF setup including sample water vial, lens and filters. (**b**) Snapshot of the setup including the GFP and the Arduino board connectivity.

arsenic concentrations, in order to exactly determine the concentration of arsenic in the sample. In Fig. 4a the GFP-based arsenic detection setup including sample water vial, lens and filters are shown, whereas Fig. 4b illustrates the Arduino board connectivity at the back side of the GFP setup.

6.2. Aqua-fruit chamber setup

In the proposed CRCD framework, in order to obtain a water sample of test fruit, the aqua-fruit chamber setup is employed, which is shown in Fig. 5. Precisely, in this setup, the sample fruit is dipped in water, which is circulated continuously with the help of a water regulatory pump with a push button (On/Off switch) mount assembly. After dipping the fruit in water for specific amount of time, the water samples of input fruit is collected in the vial shown in Fig. 5 for further processing. More precisely, Fig. 5a shows the water pump mount assembly and Fig. 5b indicates the push button setup in the proposed framework.

6.3. Display panel setup

Fig. 6a shows the front view of the display panel setup, which is used to display the results of the experiments in terms of chemical contamination measurements. Specifically, the display panel is fixed inside a metallic compartment in order to avoid external damages due to dust, heat and water. The display panel is also equipped with a power button, pump regulatory switch and LED switches shown in Fig. 6a. Fig. 6b. indicates the side

(a) **(b)**

Fig. 5. (**a**) Aqua-chamber setup with pump mount. (**b**) Push button assembly.

view of the display panel along with the aqua chamber setup mounted on it, in which slight air ventilation is also provided for experimentation purpose. More specifically, in the proposed framework, standard HD44780 LCD is used as the display panel for displaying the outputs, which is 16 characters wide with 2 rows, and displays white text on blue background. It includes a connection port of 0.1 inch pitch, single row for easy bread-boarding and wiring and also all the pins are documented on the back of the LCD to assist in wiring it up to other modules of the setup.

6.4. Main controller setup

The main controller of the experimental setup acts like a heart of the system and consists of an Arduino board. This controller interacts with all the modules of the setup including the display panel, the GFP setup, the aqua chamber and the sensor camera module. Therefore, the connectivity of the Arduino plays a major role in determining the performance of the proposed research work. Fig. 7 shows the main controller setup including the connectivity of the Arduino board with the display panel, the aqua fruit chamber and the GFP setup. Further, in the proposed CRCD framework, ArduCam MT9 MP-CMOS infrared camera module with adapter board is utilized to capture the images of sample fruits in terms of different views and various dimensions. Precisely, this camera module is placed inside an outer cabinet in order to enable clear capturing. The resolution of the camera is 1280×1024 SXGA at 30 fps, whereas the ADC is 10 bits.

(a)

(b)

Fig. 6. (**a**) Front view of the display panel. (**b**) Side view of the display panel along with the aqua chamber setup.

7. Methods and results

7.1. Chemical ripening detection using visual features

In this section, the chemical ripening detection results computed using image processing module of proposed CRCD framework are discussed in detail. Precisely, Fig. 8 shows the snapshot of the GUI named *Artificial Ripening Detector for Banana Fruit* which supports both the web portal interface as well as the mobile application interface for its processing. More precisely, Fig. 8 shows the web portal, used in proposed system, for uploading the banana pictures for processing.

Once the sample fruit image is uploaded in to the server, the proposed CRCD framework proceeds with the next step, in which checking of the input image for ripened banana/unripened banana is implemented. In the proposed framework, it employs an Artificial Neural Network (ANN), trained with more than 600 images of ripened and unripened bananas, in order to proceed with the decision process in terms of probability scores. Specifically, if the probability score for ripened banana is more than 0.9, then

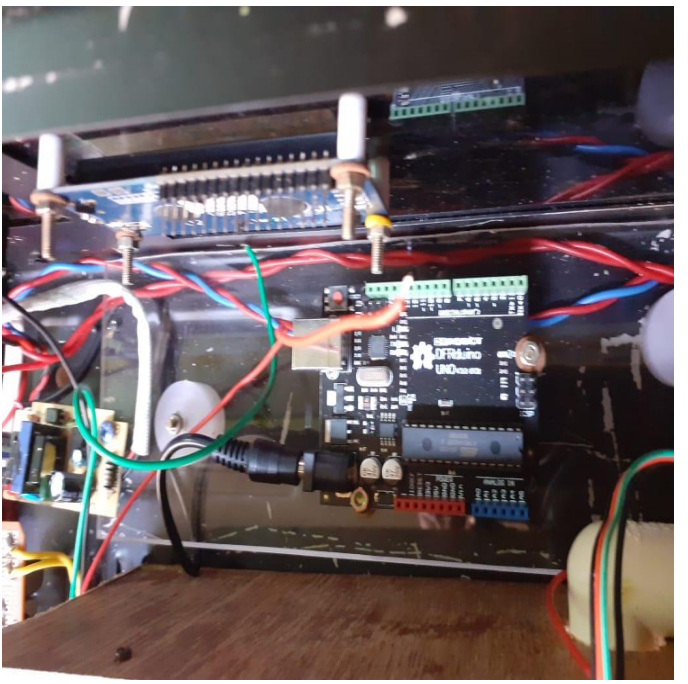


Fig. 7. Connectivity of the Arduino with the display panel, the aqua chamber and the GFP setup.

the system displays the result, as shown in Fig. 9, and proceeds for further processing. Otherwise, it displays an error message.

Fig. 10 indicates the various steps involved in the proposed CRCD framework, including gray scale conversion, noise reduction followed by the edge as well as shape detection functionalities. Specifically, in the proposed CRCD framework, Canny edge detection algorithm is employed for extracting the features. After this step, two histograms are calculated from sample fruit image namely, luminance and RGB curves. More specifically, in Fig. 11 the first histogram corresponding to the luminance values as well as the second histogram representing the RGB curves for the input image are shown. Then the resultant histograms are compared along with the respective feature descriptors of input images stored in the database. Precisely, in the proposed framework, after the forming of the histograms, the resultant graphs of input fruit image are compared against the database images. More precisely, an ANN, which is trained using the histograms having

Fig. 8. Snapshot of the web portal GUI for uploading the sample fruit image.

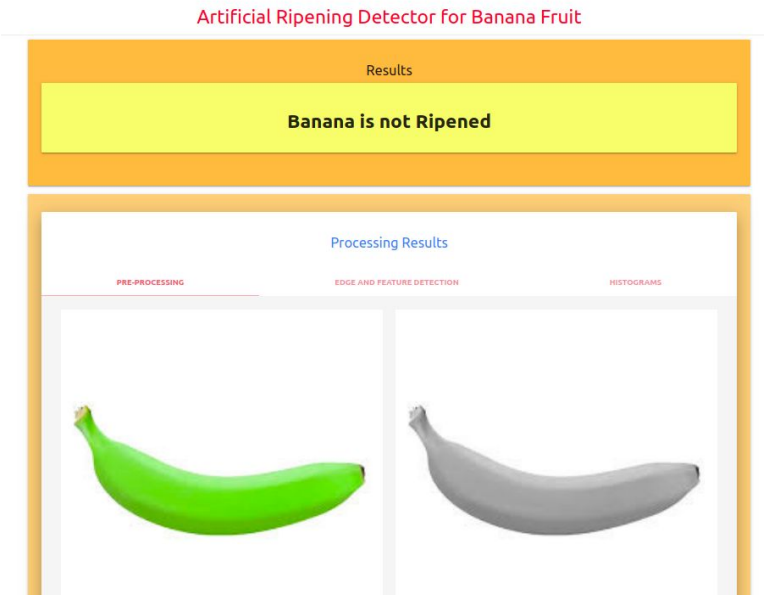


Fig. 9. Ripened vs. unripened: detecting unripened fruits.

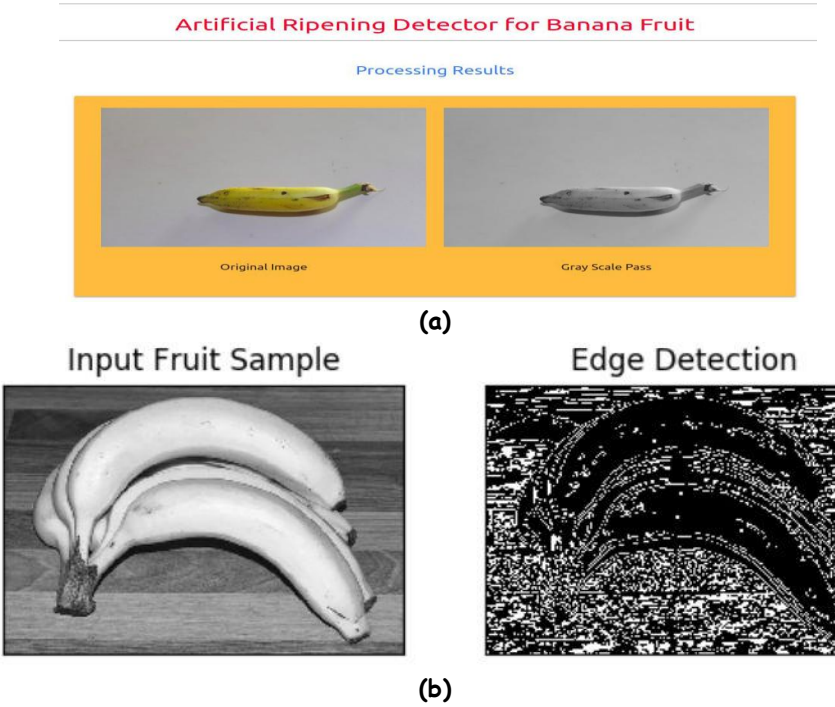


Fig. 10. (**a**) Gray scale conversion, (**b**) edge and shape detection of the input fruit image.

data size of more than 600 banana images, determines the ripening category of the sample input image. The confidence level measurements are evaluated for both the naturally ripened as well as chemically ripened categories, in order to predict exactly, whether the sample fruit is artificially ripened or not.

7.2. Chemical ripening detection results

Fig. 12a shows the snapshot of output prediction results in terms of confidence levels of 83.16% for artificially ripened banana fruit whereas Fig. 12b indicates the snapshot of output results in terms of confidence levels of 81.14% for naturally ripened banana fruit. After the completion of detection results, the back end results can be transferred to the application front end, and also to the web portal, so that the end user can view the results. Further, the mobile application interface version of the proposed CRCD framework is shown in Fig. 13, which demonstrates the performance of the proposed system in terms of displaying the results on the mobile application.

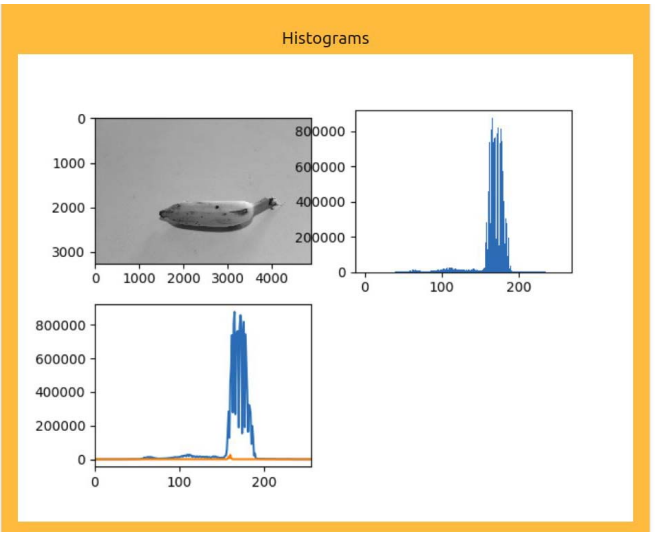


Fig. 11. Histogram graphs of the sample input fruit image.

7.3. Spectral analysis and chemical contamination computation

In the second module of the proposed CRCD framework, the NIR spectra of sample fruit image is computed and prediction results are calculated by comparing against the pre-processed spectra signatures. Specifically, for the purpose of NIR data analysis, a total of 12 readings are considered for both naturally as well as artificially ripened mango fruit samples. Further, as a preliminary classification step, principal component analysis (PCA) is performed on the selected spectra and different principal components are plotted in order to indicate the groups of samples based on their varieties. Specifically, the signature spectrum of database fruits is computed using PCA approach, before the analysis of samples, in order to reduce the noise effects as well as to obtain the better representation of the data. Fig. 14 represents the pre-processed spectra considered in the proposed framework in terms of the NIR analysis of sample mango fruits. Fig. 15 presents the signature spectra of naturally ripened mango fruit in which huge variations can be observed in the range of 600-640 nm as well as 700-800 nm wavelength measurements. In Fig. 16 the spectral signature of artificially ripened mango fruit is indicated, in which more variations are visible in the range of 700-750 nm wavelengths and thereby clear classification of ripening type of fruits can be achieved. The spectral results are further analysed with PCA, which clearly indicates that the naturally and artificially ripened mango samples spectra are falling in different wavelength segments. Further,

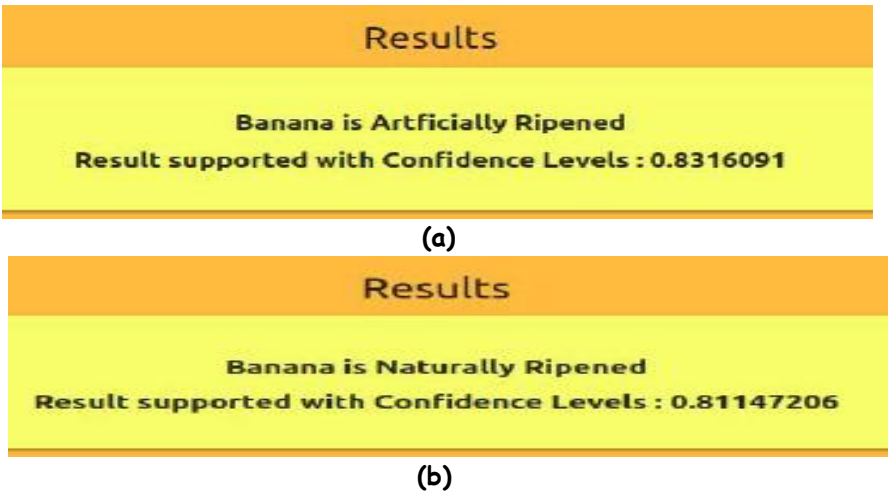


Fig. 12. (**a**) Final detection results for artificially ripened banana fruit. (**b**) Detection results for naturally ripened banana fruit.

Fig. 13. Snapshots of artificial vs. natural ripening detection results displayed on mobile application interface.

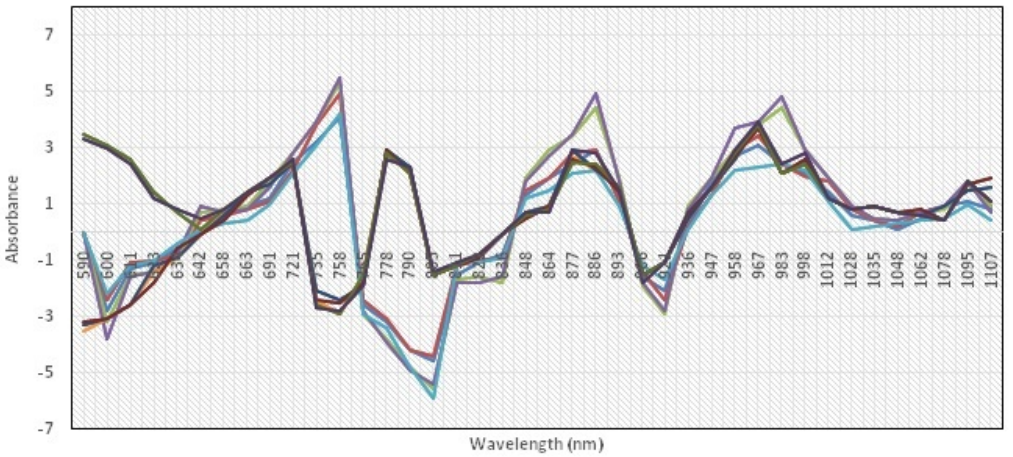


Fig. 14. Pre-processed spectral signature computed for a sample mango fruit.

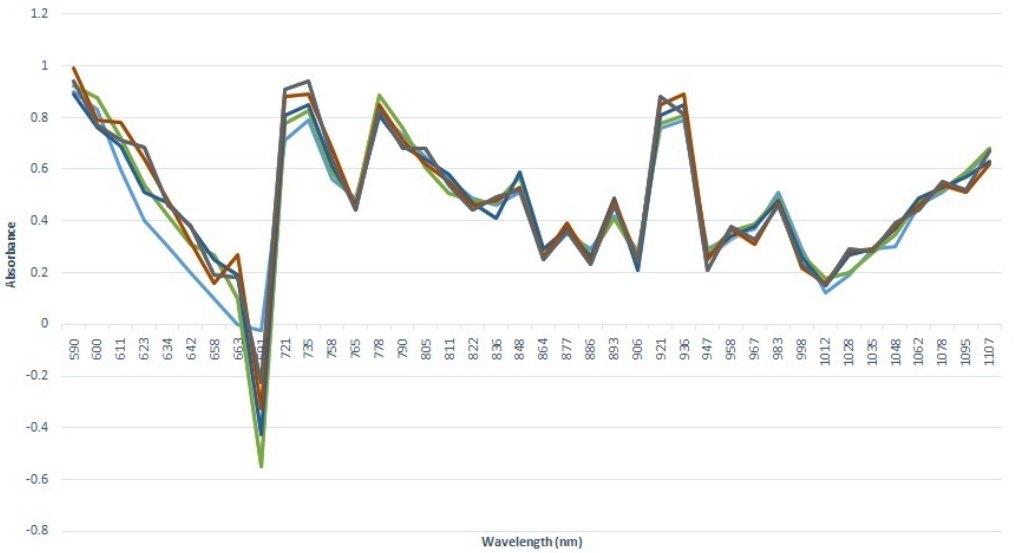


Fig. 15. Spectral signature of a naturally ripened mango fruit.

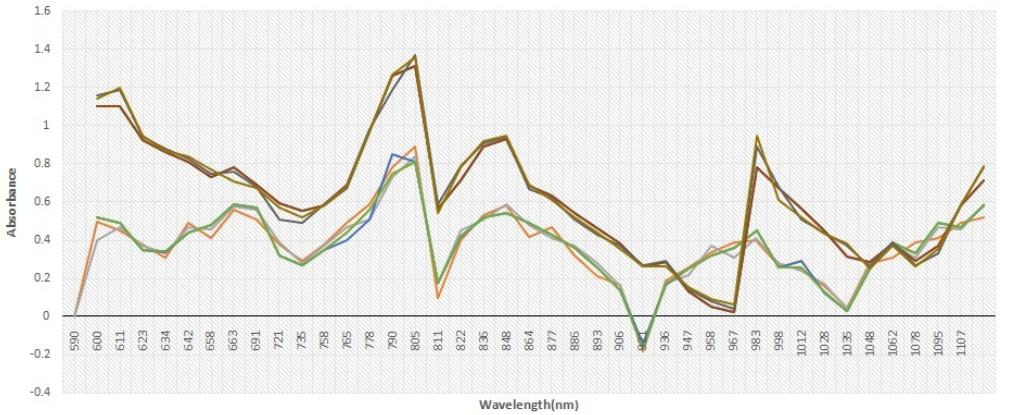


Fig. 16. Spectral signature of an artificially ripened mango fruit.

the resultant spectra graphs also indicate significant amount of variations in the spectral signatures of both fruits and thereby demonstrate the performance of the proposed CRCD framework in terms of achieving accurate prediction results for calcium carbide based artificially ripened mango fruits.

7.4. Chemical contaminations quantification and results

In the proposed CRCD framework, in order to quantify the amount of chemical contaminations in terms of arsenic contents, five different datasets consisting each of 30 fruit samples are considered as given below:

S1: Fruit samples treated with 2% calcium carbide,

S2: Fruit samples treated with 4% calcium carbide,

S3: Fruit samples stored in closed container with 1% calcium carbide,

S4: Fruit samples stored in closed container with 3% calcium carbide and

S5: Naturally ripened fruits.

Specifically, in order to confirm the presence of arsenic content in the artificially ripened fruit samples, the datasets are analysed with turbidity measurements. More specifically, Table 1 indicates the arsenic content present in chemically ripened fruits in terms of measurements in ng/g. Specifically, ng/g is equivalent to 1 ppb (parts per billion). The presence of arsenic content is ranging from 0 to 290 ng/g, in which naturally ripened fruit samples show the maximum arsenic content of 8 ng/g, whereas the largest amount of arsenic presence is observed in calcium carbide treated fruit samples, direct consumption of which is highly dangerous to humans.

Tab. 1. Presence of arsenic content in terms of levels in ng/g for different datasets: S1, S2, S3, S4 and S5, respectively.

SET	0-30 ng/g	30-70 ng/g	70-290 ng/g
S1	27	3	-
S2	20	10	-
S3	4	24	2
S4	-	-	30
S5	30	-	-

Fig. 17. Result display showing arsenic chemical composition.

Fig. 17 shows the output in the front panel display of the proposed CRCD framework, in which arsenic composition present in the input fruit is displayed. Specifically, arsenic composition is 78.23% for the sample input fruit, as shown in Fig. 17, which is higher than the prescribed threshold limits, hence it suggests that it is harmful to consume the given sample fruit.

The performance of the proposed CRCD framework is also evaluated in terms of sensitivity and specificity measures by means of comparisons with ground truth values.

Precisely, the 200 sample fruits consisting of both natural and artificially ripened categories are considered for evaluating the performance of the proposed framework. More precisely, sensitivity of the CRCD framework defining the probability of positive result for the given set of artificially ripened fruits is indicated as 91.25%. Further, the specificity of the proposed CRCD framework is also computed as 80.25%, which defines the probability of negative result for the given set of naturally ripened fruits. In this way, the reasonable rates of sensitivity as well as specificity rates demonstrate that the proposed CRCD framework is reliable and hence it can be employed in real-time chemical ripening detection systems.

8. Conclusion and future work

In this paper, a new chemical ripening and contamination detection framework is introduced, which utilizes both visual as well as infrared spectrometric features. The experiments conducted on both the software-based and hardware-based setups clearly demonstrate the efficiency of the proposed framework in terms of confidence levels followed by the measurement of presence of arsenic in the sample fruit.

In future, the proposed framework can be extended as real-time detection tool for analyzing other types of fruits including papaya, tomato and other species.

Acknowledgement

This research work is supported by Karnataka Science and Technology Promotion Society (KSTePS) under VGST research grant no. KSTePS/VGST/RGS/F/2017-18/GRD-726. of Government of India.

References

[1] S. Ahmad, A. K. Thompson, I. A. Hafiz, and A. A. Asi. Effect of temperature on the ripening behavior and quality of banana fruit. *International Journal of Agriculture and Biology*, 3(2):224–227, 2001. `http://www.fspublishers.org/published_papers/87164_..pdf`.

[2] S. Ansari and S. Salankar. An overview on thermal image processing. In V. K. Solanki, V. B. Semwal, R. González-Crespo, and V. Bijalwan, editors, *Proc. 2nd Int. Conf. Research in Intelligent and Computing in Engineering RICE 2017*, volume 10 of *Annals of Computer Science and Information Systems*, pages 117–120, Gopeshwar, Uttrakhand, India, 24-26 Mar 2017. Polish Information Processing Society, Warsaw. doi:10.15439/2017R111.

[3] A. A. Bhosale and K. K. Sundaram. Nondestructive method for ripening prediction of papaya. *Procedia Technology*, 19:623–630, 2015. Part of special issue: L. Moldovan, editor, *Proc. 8th Int. Conf. Interdisciplinarity in Engineering, INTER-ENG 2014*, 9-10 Oct 2014, Tirgu Mures, Romania. doi:10.1016/j.protcy.2015.02.088.

[4] F. J. García-Ramos, C. Valero, I. Homer, et al. Non-destructive fruit firmness sensors: A review. *Spanish Journal of Agricultural Research*, 3(1):61–73, 2005. doi:10.5424/sjar/2005031-125.

[5] V. Hallur, B. Atharga, A. Hosur, et al. Design and development of a portable instrument for the detection of artificial ripening of banana fruit. In *Int. Conf. Circuits, Communication, Control and Computing*, pages 139–140, Bangalore, India, 21-22 Nov 2014. IEEE, 2015. doi:10.1109/CIMCA.2014.7057776.

[6] M. C. House, M. Nelson, and E. S. Haber. The vitamin A, B, and C content of artificially versus naturally ripened tomatoes. *Journal of Biological Chemistry*, 81(3):495–504, 1929. doi:10.1016/S0021-9258(18)63704-4.

[7] R. Karthika, K. V. M. Ragadevi, and N. Asvini. Detection of artificially ripened fruits using image processing. *Internatinal Journal of Advanced Science and Engineering Research*, 2(1):576–582, 2017. `http://www.ijaser.in/journals/view/volume2/issue1/detection-of-artificially-ripened-fruits-using/221`.

[8] J. Kathirvelan and R. Vijayaraghavan. An infrared based sensor system for the detection of ethylene for the discrimination of fruit ripening. *Infrared Physics & Technology*, 85:403–409, 2017. doi:10.1016/j.infrared.2017.07.022.

[9] S. Kothari and H. Channe. Detection of nutrients and chemicals in food products using sensors in smart phones. *Internatinal Journal of Engineering and Computer Science*, 4(4):11651–11652, 2015. `https://www.ijecs.in/index.php/ijecs/article/view/1696`.

[10] A. J. Lakade, K. Sundar, and P. H. Shetty. Gold nanoparticle-based method for detection of calcium carbide in artificially ripened mangoes *(Magnifera indica)*. *Food Additives & Contaminants: Part A*, 35(6):1078–1084, 2018. doi:10.1080/19440049.2018.1449969.

[11] A. J. Lakade, Venkataraman V., R. Ramasamy, and P. H. Shetty. NIR spectroscopic method for the detection of calcium carbide in artificial ripening of mangoes *(Magnifera indica)*. *Food Additives & Contaminants: Part A*, 36(7):989–995, 2019. doi:10.1080/19440049.2019.1605206.

[12] V. H.-C. Liao and K.-L. Ou. Development and testing of green fluorescent protein-based bacterial biosensor for measuring bioavailable arsenic in contaminated groundwater samples. *Environmental Toxicology and Chemistry*, 24(7):1624–1631, 2005. doi:10.1897/04-500R.1.

[13] S. Maheswaran, S. Sathesh, P. Priyadarshini, and B. Vivek. Identification of artificially ripened fruits using smart phones. In *2017 Int. Conf. Intelligent Computing and Control I2C2*, pages 1–6, Coimbatore, India, 23-24 Jun 2017. IEEE, 2018. doi:10.1109/I2C2.2017.8321857.

[14] M. R. Meghana, R. Roopalakshmi, T. E. Nischitha, and P. Kumar. Detection of chemically ripened fruits based on visual features and non-destructive sensor techniques. In D. Pandiana, X. Fernando, Z. Baig, and F. Shi, editors, *Proc. Int. Conf. ISMAC in Computational Vision and Bio-Engineering ISMAC-CVB 2018*, volume 30 of *Lecture Notes in Computational Vision and Biomechanics*, pages 865–872, Palladam, India, 16-17 May 2018. Springer, Cham 2019. doi:10.1007/978-3-030-00665-5_84.

[15] P. P. Ray, S. Pradhan, R. K. Sharma, et al. IoT based fruit quality measurement system. In *2016 Online Int. Conf. Green Engineering and Technologies IC-GET*, pages 224–229, Coimbatore, India, 19 Nov 2016. IEEE, 2017. doi:10.1109/GET.2016.7916620.

[16] R. Roopalakshmi, C. Shastri, P. Hegde, et al. Neural networks-based framework for detecting chemically ripened banana fruits. In H. Sharma, A. Pundir, N. Yadav, et al., editors, *Recent Trends in Communication and Intelligent Systems – Proc. Int. Conf. Recent Trends in Communication & Intelligent Systems ICRTCIS 2019*, volume of *Algorithms for Intelligent Systems*, pages 55–61, Jaipur, India, 8-9 Jun 2019. Springer, Singapore 2020. doi:10.1007/978-981-15-0426-6_6.

[17] R. P. Salunkhe and A. A. Pathil. Image processing for mango ripening stage detection: RGB and HSV method. In *2015 3rd Int. Conf. Image Information Processing ICIIP*, pages 362–365, Waknaghat, India, 21-24 Dec 2015. IEEE, 2016. doi:10.1109/ICIIP.2015.7414796.

[18] V. Srividhya, K. Sujatha, and R. S. Ponmagal. Ethylene gas measurement for ripening of fruits using image processing. *Indian Journal of Science and Technology*, 9(31):1–7, 2016. doi:10.17485/ijst/2016/v9i31/93838.

[19] tensorflower gardener and mingxingtan (GitHub nicknames). Running Inception on Cloud TPU, 2021. `https://cloud.google.com/tpu/docs/tutorials/inception`. [Last accessed Jun 2021].

[20] A. Verma, R. Hegadi, and K. Sahu. Development of an effective system for remote monitoring of banana ripening process. In *2015 IEEE Int. WIE Conf. Electrical and Computer Engineering WIECON-ECE*, pages 534–537, Dhaka, Bangladesh, 19-20 Dec 2015. IEEE, 2016. doi:10.1109/WIECON-ECE.2015.7443987.

Roopalakshmi R. Dr. Roopalakshmi R. is currently working as Associate Professor in the Computer Science & Engineering (CSE) department of Manipal Institute of Technology (MIT), MAHE, Manipal. She has completed her M.Tech. (CSE) from P.E.S. Institute of Technology (PESIT) – Bangalore. Followed by her post graduation, she has pursued her Full-time Ph.D. in the Information Technology Department of the National Institute of Technology Karnataka (NITK) – Surathkal, which is one of the most prestigious and oldest NITs of India. She has published a good number of journal papers in very high quality journals such as Signal Processing, Elsevier, and technically sound enough for having her paper presentations in Tier-I International Conferences, and henceforth her research articles are widely available in digital libraries such as IEEE Xplore, ACM, Elsevier science direct and Springer.
She proved to be a Best Researcher, by involving herself as Principal Investigator in Funded research projects such as DST-Women Scientist Research Grant, worth Rs. 18.20 Lakhs from MHRD of India, New Delhi, and VGST research grant scheme of worth Rs. 5.00 Lakhs from Karnataka Science and Technology Promotion Society (KSTePS), Govt. of Karnataka.

Nonlinear Evolutionary PDE-Based Refinement of Optical Flow

Hirak Doshi and Nori Uday Kiran
Department of Mathematics and Computer Science
Sri Sathya Sai Institute of Higher Learning, Puttaparthi, India
hirakdoshi,nudaykiran@sssihl.edu.in

Abstract. The goal of this paper is to propose two nonlinear variational models for obtaining a refined motion estimation from an image sequence. Both the proposed models can be considered as a part of a generalized framework for an accurate estimation of physics-based flow fields such as rotational and fluid flow. The first model is novel in the sense that it is divided into two phases: the first phase obtains a crude estimate of the optical flow and then the second phase refines this estimate using additional constraints. The correctness of this model is proved using an evolutionary PDE approach. The second model achieves the same refinement as the first model, but in a standard manner, using a single functional. A special feature of our models is that they permit us to provide efficient numerical implementations through the first-order primal-dual Chambolle-Pock scheme. Both the models are compared in the context of accurate estimation of angle by performing an anisotropic regularization of the divergence and curl of the flow respectively. We observe that, although both the models obtain the same level of accuracy, the two-phase model is more efficient. In fact, we empirically demonstrate that the single-phase and the two-phase models have convergence rates of order $O(1/N^2)$ and $O(1/N)$ respectively.

Key words: optical flow, evolutionary PDE, variational methods, primal-dual, convergence

1. Introduction

Optical flow plays a key role in many advanced Computer Vision applications. It is a rich source of information on perceptible motion in our visual world. It's reliable estimation is thus important and at the same time challenging. Assuming the principle of local conservation of intensity and small temporal variations, optical flow involves the recovery of a function $\mathbf{u} = (u, v)$ such that

$$f(\mathbf{x}, \tau) = f(\mathbf{x} + \mathbf{u}, \tau + \Delta\tau)\,,$$

where $f : \Omega \times [0, T] \to \mathbb{R}$ is the image sequence, $\Omega \subset \mathbb{R}^2$ is open and bounded. This establishes a correspondence between pixel motions. Using first-order approximations the above relation can be written as

$$f_\tau + \nabla f \cdot \mathbf{u} = 0\,, \tag{1}$$

Machine GRAPHICS & VISION 30(1/4):45–65, 2021. DOI: 10.22630/MGV.2021.30.1.3 .

which is widely known as the Optical Flow Constraint (OFC). To recover the velocity components using a variational minimization approach one writes (1) as

$$\min_{\mathbf{u}} J_1(\mathbf{u}) = \int_\Omega (f_t + \nabla f \cdot \mathbf{u})^2 \,.$$

This is the simplest least-square minimization. This problem is ill-posed as it leads to the aperture problem. Additional regularization terms are necessary to ensure well-posedness. The most common regularization term is the quadratic smoothness penalizing the gradient of the components of the flow originally introduced by Horn and Schunck [16] in their seminal work. Cohen [8] and Kumar et al. [17] used the L^1 regularization which is more robust to outliers and preserves important edge information. A new discontinuity preserving optical flow model with L^1 norm on the OFC was proposed and studied by Aubert et al. [1] in the space of functions of bounded variations $BV(\Omega) \times BV(\Omega)$. The well-posedness of the Horn and Schunck model, as well as the Nagel model was studied by Schnörr [23] in the space $H^1(\Omega) \times H^1(\Omega)$. Taking a step further, the authors in [5] proposed a $L^p - TV/L^p$ ($p = 1$ or 2) model combining both L^1 and L^2 terms. The behaviour of their regularization term is similar to the Huber function:

$$H(x;\epsilon) = \begin{cases} \frac{x^2}{2\epsilon}, & 0 \le |x| \le \epsilon \\ |x| - \frac{\epsilon}{2}, & |x| > \epsilon \,. \end{cases}$$

A detailed review and rigorous analysis of several variational optical flow models within the framework of calculus of variations can be found in [15].

Though most of the estimation involving rigid or quasi-rigid motion can be handled by minimizing OFC with a suitable regularization, it is insufficient to provide an accurate estimation for fluid-based images. Traditional computer vision techniques may not be suitable to capture these deformations of brightness patterns because of the high spatio-temporal turbulence in these sequences. These reasons have motivated researchers to look for an alternative constraint that can not only preserve pixel-correspondence but also capture certain intrinsic features of the flow. This paradigm shift hints at constraints that are physics-dependent. A lot of work has been done involving physics-based constraints for fluid motion estimation [9, 10, 18, 19, 20]. In [12], we have proposed a constraint-based refinement of optical flow. Using an image-driven evolutionary PDE model resulting from a quadratic regularization we have shown the well-posedness of such a refinement principle. An important characteristic of the model is the possibility of a diagonalization by the Cauchy-Riemann operator leading to a decoupled system involving diffusion of the curl and a multiplicative perturbation of the laplacian of the divergence of the flow. For a specific case, it was shown that the model is close to the physics-based model [18] using a modified augmented Lagrangian method.

The current work proposes a unified framework for a nonlinear evolutionary PDE-based refinement of optical flow. The first model is a two-phase refinement process.

A crude pixel correspondence is obtained in the first phase which constitutes a good starting solution. In the next phase, this estimate is refined using additional constraints. This constraint is chosen motivated by the non-conservation term $f\nabla \cdot \mathbf{u}$ in the physics-based constraint, (see [12]). The second model estimates the flow directly in a single phase. In this case the additional constraint is chosen motivated by the harmonic constraint-based regularization (see [30]). This approach replaces the oriented smoothness constraint with a weighted decomposition of divergence and curl of flow. We aim to capture the rotational features better by preserving edge information and improving the accuracy of the flow. Thus we consider only the curl component in our framework with an anisotropic weight term.

The total variation regularization leads to Δ_1, the 1-Laplacian operator in the Euler-Lagrange equations. Obtaining a stable convergent scheme is a difficult task because of the singularity of the operator at the origin. As a result most of the implementation methods often yield slower algorithms. In this direction important contributions were made by Chambolle [6] and Zach [27]. Chambolle and Pock [7] proposed a first-order primal-dual algorithm for solving non-smooth convex optimization problems. This further opened up newer directions as a large class of problems in Image Processing could be solved within this framework. In our work, we use the Chambolle-Pock algorithm for both of our models. The numerical implementation of the algorithm has two main steps, namely updating the primal variables by solving a system of equations at each iteration and updating the dual variables by computing the point-wise projection maps onto the unit ball. Both of these steps are computationally expensive. As a result the second model yields a slower algorithm with a convergence rate of order $O(1/N^2)$. By $O(1/N^2)$ we mean if ϵ is the error threshold then the number of iterations required to reach this threshold is $1/\epsilon^2$. The first model splits the above-mentioned steps in two-phases. This leads to a faster algorithm with a convergence rate of $O(1/N)$.

The organization of the paper is as follows. In Section 2 we give the general formulation and describe our model in detail. Next in Section 3, we study the mathematical well-posedness of our formulation using an evolutionary PDE approach. Subsequently, we employ the first-order primal-dual Chambolle-Pock algorithm to our models and derive the necessary optimality conditions in Section 4. We then discuss the implementation details, discretization of our models and empirically demonstrate the nature of convergence in Section 5.

2. Our Model Description

Our general formulation is given as:

$$J(\mathbf{u}) = \int_\Omega \rho(|f_t + \nabla f \cdot \mathbf{u}|) + \alpha \sum_{i=1}^{2} \int_\Omega \gamma(|\nabla u_i|) + \beta \int_\Omega \phi(x, f, \nabla f)\psi(\mathbf{u}, \nabla \mathbf{u})\,, \qquad (2)$$

where $\rho : \mathbb{R} \to \mathbb{R}$ is a function of the optical flow constraint, $\gamma : \mathbb{R} \to \mathbb{R}$ governs the regularization of the flow. The functions ϕ and ψ are chosen specific to applications. A summary of some of the variational models which belong to this framework is listed in Table 1. Based on the choice functions mentioned in this table, the constraint-based refinement formulation becomes

$$(M1) \quad J(\mathbf{u}) = \alpha \sum_{i=1}^{2} \int_{\Omega} |\nabla u_i| + \beta \int_{\Omega} f^2 (\nabla \cdot \mathbf{u})^2 \,, \tag{3}$$

where $|\cdot|$ is the Euclidean norm. Starting with $\mathbf{u} = \mathbf{u}_0$, where $\mathbf{u}_0$ is the Horn and Schunck optical flow the above formulation obtains a refinement of $\mathbf{u}_0$ driven by the additional constraint $f\nabla \cdot \mathbf{u}$. This term is the non-conservation term in the physics-based constraint due to non null-out of plane components [14]. This constraint preserves the spatial characteristics and vorticities of the flow. Thus this model can extract flow information from fluid-based digital imagery much better.

From the choice functions, our second formulation is given as:

$$(M2) \quad J(\mathbf{u}) = \int_{\Omega} (f_t + \nabla f \cdot \mathbf{u})^2 + \alpha \sum_{i=1}^{2} \int_{\Omega} |\nabla u_i| + \beta \int_{\Omega} \frac{\lambda^2}{|\nabla f|^2 + \lambda^2} (\nabla_H \cdot \mathbf{u})^2 \,. \tag{4}$$

The additional constraint in this formulation is motivated by the harmonic-constraint based regularization discussed in [30]. By associating an anisotropic weight term with the curl in our formulation there are two main advantages. First, we are able to get a precise estimation of the infinitesimal rotation within the regions. Secondly, we achieve a better alignment of small vectors in the flow. This leads to an overall improvement in the endpoint error.

In either case, the choice of ϕ decides the influence of the image term in the regularization process. If $\phi = 1$, then the additional constraint term in the functional is

Tab. 1. Some choices for the functions.

	$\rho(x)$	$\gamma(x)$	$\phi(x, f, \nabla f)$	$\psi(\mathbf{u}, \nabla \mathbf{u})$
Horn and Schunck [16]	x^2	x^2	0	0
Cohen [8]	x^2	x	0	0
Aubert [1]	x	$\sqrt{1+x^2}$	$c(x)$	$\mathbf{u}^2$
$L^1 - TV$ [27]	x	x	0	0
Our Model (M1)	0	x	f^2	$(\nabla \cdot \mathbf{u})^2$
Our Model (M2)	x^2	x	$\frac{\lambda^2}{\|\nabla f\|^2 + \lambda^2}$	$(\nabla_H \cdot \mathbf{u})^2$

flow-driven, i.e. independent of the influence of the image data. The weight parameters α, β play an important role in the regularization process. For rigid-body like motion which requires important edge-information to be preserved a higher value of α is preferred. For fluid-based images where there is less edge-prominence, we choose a higher β value.

3. Well-Posedness

In this section we discuss the mathematical well-posedness of the proposed formulation. Let us denote $\mathbf{u} = (u_1, u_2)$. The space $W^{1,p}(\Omega), p > 1$, is the reflexive Banach space

$$W^{1,p}(\Omega) = \{u \in L^p(\Omega) : D^\alpha u \in L^p(\Omega), |\alpha| \leq 1\}$$

with the usual norm

$$\|u\|_{W^{1,p}} = \left(\sum_{|\alpha| \leq 1} \|D^\alpha u\|_{L^p}^p \right)^{1/p}, 1 \leq p < \infty \,.$$

To study the mathematical well-posedness of the proposed formulation we consider the following approximation.

$$J_{p,\mathrm{R}}(\mathbf{u}) = \beta \int_\Omega \phi(x, f, \nabla f)(\nabla \cdot \mathbf{u})^2 + \frac{\alpha}{p} \int_\Omega \{|\nabla u_1|^p + |\nabla u_2|^p\}, \quad 1 < p < 2 \,. \tag{5}$$

This functional being strictly convex in $W^{1,p}(\Omega)$ admits a unique minimizer. For this discussion we consider $\phi(x, f, \nabla f) = f^2$. The first important step is to show that $J_{p,\mathbf{R}}$ converges to $J_{1,\mathbf{R}}$ as $p \to 1$. For this, we refer to the discussion in Section 3.4 in [21].

Lemma 1.

$$\lim_{p \to 1} \frac{1}{p} \int_\Omega |\nabla u|^p = \int_\Omega |\nabla u| \,. \tag{6}$$

Remark 1. As $J_{p,\mathrm{R}}(\mathbf{u}) \to J_{1,\mathrm{R}}(\mathbf{u})$, $p \to 1$, the corresponding Euler-Lagrange equations $A_p = \Delta_p$ associated with the regularization term also converges to $A_1 = \Delta_1$.

Remark 2. The case $p = 2$ leads to a linear diffusion-driven refinement process. We have previously studied and discussed this case in [12].

The associated parabolic system corresponding to the Euler-Lagrange equations of

(5) are given as

$$\begin{cases} \dfrac{\partial u_1}{\partial t} = \Delta_p u_1 + a_0 \dfrac{\partial}{\partial x}[f^2((u_1)_x + (u_2)_y)] \text{ in } \Omega \times (0,\infty)\,, \\ \dfrac{\partial u_2}{\partial t} = \Delta_p u_2 + a_0 \dfrac{\partial}{\partial y}[f^2((u_1)_x + (u_2)_y)] \text{ in } \Omega \times (0,\infty)\,, \\ u_1(x,y,0) = u_1^0 \text{ in } \Omega\,, \\ u_2(x,y,0) = u_2^0 \text{ in } \Omega\,, \\ u_1 = 0 \text{ on } \partial\Omega \times (0,\infty)\,, \\ u_2 = 0 \text{ on } \partial\Omega \times (0,\infty)\,, \end{cases} \tag{7}$$

where (u_1^0, u_2^0) is the starting feasible solution obtained by the Horn and Schunck optical flow, $a_0 = 2\beta/\alpha$. Rewriting the system in an abstract form leads us to

$$\begin{cases} \dfrac{d\mathbf{u}}{dt} + \mathcal{A}_p \mathbf{u} = 0, \quad t > 0\,, \\ \mathbf{u}(0) = \mathbf{u}_0 \in H^1(\Omega)^2\,. \end{cases} \tag{8}$$

Here the operator $\mathcal{A}_p = A_p + F$, where

$$A_p \mathbf{u} = -\begin{bmatrix} \Delta_p u_1 \\ \Delta_p u_2 \end{bmatrix}, \qquad F\mathbf{u} = -a_0 \begin{bmatrix} \dfrac{\partial}{\partial x}[f^2((u_1)_x + (u_2)_y)] \\ \dfrac{\partial}{\partial y}[f^2((u_1)_x + (u_2)_y)] \end{bmatrix}.$$

We will show that both the operators A_p and F are maximal monotone in $W^{1,p}(\Omega) \cap L^2(\Omega)$ and $L^2(\Omega)$, respectively.

Lemma 2. The operators A_p and F is maximal monotone in $W^{1,p}(\Omega) \cap L^2(\Omega)$ and $L^2(\Omega)$, respectively.

Proof. The maximal monotonicity of A_p follows directly from the discussions in [26]. To show monotonicity we show that $\langle F\mathbf{u}, \mathbf{u}\rangle \geq 0$. Indeed,

$$\begin{aligned} \langle F\mathbf{u}, \mathbf{u}\rangle &= -a_0 \int_\Omega \left\{ \frac{\partial}{\partial x}[f^2((u_1)_x + (u_2)_y)]u + \frac{\partial}{\partial y}[f^2((u_1)_x + (u_2)_y)]v \right\}, \\ &= a_0 \int_\Omega \{f^2((u_1)_x + (u_2)_y)u_x + f^2((u_1)_x + (u_2)_y)v_y\}\,, \end{aligned}$$

$$= a_0 \int_\Omega f^2((u_1)_x + (u_2)_y)^2 \geq 0\,,$$

proving the monotonicity of F. To show the maximality we have to show that

$$\text{Ran}(I + F) = L^2(\Omega)^2\,, \tag{9}$$

i.e. there exists $\mathbf{u}$ for all $\mathbf{f} \in L^2(\Omega)^2$ such that $\mathbf{u} + F\mathbf{u} = \mathbf{f}$ holds. Let $\mathbf{f} = (f, g) \in L^2(\Omega)^2$ and consider the system

$$u + \frac{\partial}{\partial x}[f^2((u_1)_x + (u_2)_y)] = f\,,$$

$$v + \frac{\partial}{\partial y}[f^2((u_1)_x + (u_2)_y)] = g\,,$$

where $f, g \in L^2(\Omega)$. Applying the Cauchy-Riemann operator

$$R = \begin{bmatrix} \partial_y & -\partial_x \\ \partial_x & \partial_y \end{bmatrix}$$

on both sides we obtain the decoupled system

$$(u_1)_y - (u_2)_x = f_y - g_x\,, \tag{10}$$

$$(\Delta \circ k)((u_1)_x + (u_2)_y) = f_x + g_y\,, \tag{11}$$

where k is the image-dependent multiplicative function $k : f \mapsto 1 + a_0 f^2$. The first equation (10) governs the curl of the flow $\mathbf{u} = (u_1, u_2)$. The second equation (11) indicates a non-homogeneous weighted diffusion process on the divergence with a weight k. Let us define $h_1 = f_y - g_x$ and $h_2 = f_x + g_y$. Solving the second equation gives us an expression for the divergence of the flow. Let us call this as h_3. We thus obtain the following system

$$(u_1)_y - (u_2)_x = h_1\,,$$

$$(u_1)_x + (u_2)_y = h_3/k\,.$$

These are the inhomogeneous Cauchy-Riemann equations. In a compact form we rewrite them as

$$R\mathbf{u} = \tilde{\mathbf{f}}\,,$$

where $\tilde{\mathbf{f}} = (h_1, h_3/k)$. The operator R^{-1} is a continuous operator of order -1 in the space $W^{1,p}(\Omega)$. Hence there exists a unique $\mathbf{u}$ such that $\mathbf{u} + F\mathbf{u} = \mathbf{f}$ holds. This concludes the proof. □

Now that we have shown the maximal monotonicity let us define a function $\Phi_p : L^2(\Omega)^2 \to (-\infty, +\infty]$ by

$$\Phi_p(\mathbf{u}) = \begin{cases} J_{p,\mathbf{R}}(\mathbf{u}), & \mathbf{u} \in [W^{1,p}(\Omega) \cap L^2(\Omega)]^2 \\ +\infty, & \mathbf{u} \in [L^2(\Omega) \setminus W^{1,p}(\Omega)]^2 \end{cases} .$$

Then clearly Φ_p is convex and lower semi-continuous. Also Φ_p is proper since $D(\Phi_p) = D(A_p) \cap D(F) \neq \emptyset$. Thus the associated subdifferential $\partial\Phi_p(\mathbf{u}) \equiv \mathcal{A}_p$ is maximal monotone. Thus there is a unique solution $\mathbf{u}$ of the inclusion

$$0 \in \mathbf{u}'(t) + \partial\Phi_p(\mathbf{u})$$

satisfying the initial conditions.

4. The Primal-Dual Framework

The primal-dual method is a numerical tool for solving optimization problems. The main idea is to replace a primal problem with an equivalent saddle point problem by introducing dual variables and employ efficient algorithms to obtain the desired convergence. In the recent past several saddle point frameworks have been proposed for variational problems in image processing and computer vision [7, 28, 29]. As our formulation involves non-smooth convex functionals, the most suitable framework is the one proposed by Chambolle and Pock [7]. Let $\Omega \subset \mathbb{R}^2$ be an open, bounded set, $\mathcal{X}, \mathcal{Y}$ be two finite-dimensional vector spaces with the scalar product $(\cdot, \cdot)$ and the norm $\|\cdot\|$. Denote the primal variable $\mathbf{u} = (u_1, u_2)$ and the dual variable $\mathbf{d} = (d_1, d_2, d_3)$. We first consider the variational problem in the following form

$$\arg\min_{\mathbf{u}} G(\mathbf{u}) + F(K\mathbf{u}) \,. \tag{12}$$

where $F, G : \mathcal{X} \to [0, \infty]$ are convex, proper and lower-semicontinuous functionals, $K : \mathcal{X} \to \mathcal{Y}$ is a continuous, linear operator. The equivalent primal-dual formulation is given as

$$\arg\min_{\mathbf{u}} \arg\max_{\mathbf{d}} \; G(\mathbf{u}) + (K\mathbf{u}, \mathbf{d}) - F^*(\mathbf{d}) \,, \tag{13}$$

where F^* is the convex conjugate of F. Table 2 gives a summary of each term of our model using the above notations. Given a $\tau, \sigma > 0$, an initial $(\mathbf{u}^0, \mathbf{d}^0) \in \mathcal{X} \times \mathcal{Y}$, the Chambolle-Pock algorithm solves the saddle point problem (13) by the following algorithm:

$$\mathbf{d}^{k+1} = \mathbf{prox}_{\sigma F^*}(\mathbf{d}^k + \sigma K \bar{\mathbf{u}}^k) \,,$$

$$\mathbf{u}^{k+1} = \mathbf{prox}_{\tau G}(\mathbf{u}^k - \tau K^* \mathbf{d}^{k+1}),$$
$$\bar{\mathbf{u}}_{k+1} = 2\mathbf{u}_{k+1} - \mathbf{u}_k \qquad \text{(over-relaxation)},$$

where

$$\mathbf{prox}_{\tau G}(\mathbf{d}) = (I + \tau \partial G)^{-1}(\mathbf{d}) = \arg\min_{\mathbf{u}} \left\{ \frac{1}{2}\|\mathbf{u} - \mathbf{d}\|^2 + \tau G(\mathbf{u}) \right\}$$

is the proximal or the resolvent operator. This can be thought of as a trade-off between minimizing G and being close to $\mathbf{d}$. We now employ the above algorithm for our problem and derive the necessary optimality conditions.

4.1. Optimality Condition for Our Model M1

In this case we have

$$G(\mathbf{u}) = 0, \quad F(K\mathbf{u}) = \frac{1}{2}\int_\Omega f^2 (\nabla \cdot \mathbf{u})^2 + \sum_{i=1}^{2} \int_\Omega |\nabla u_i|.$$

The Operator K is given as

$$K\mathbf{u} = \begin{bmatrix} \nabla & 0 \\ 0 & \nabla \\ f^2\partial_x & f^2\partial_y \end{bmatrix} \begin{bmatrix} u_1 \\ \\ u_2 \end{bmatrix}.$$

Therefore,

$$K^*\mathbf{d} = -\begin{bmatrix} \nabla\cdot & 0 & \partial_x(f^2\cdot) \\ 0 & \nabla\cdot & \partial_y(f^2\cdot) \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix}.$$

Tab. 2. Summary of the terms in the Primal-Dual Formulation.

Model	$\phi(x, f, \nabla f)$	$G(\mathbf{u})$	$F(K\mathbf{u})$	K
(M1)	f^2	0	$\frac{1}{2}\int_\Omega \phi(x,f,\nabla f)(\nabla\cdot\mathbf{u})^2 + \sum_{i=1}^{2}\int_\Omega \lvert\nabla u_i\rvert$	$\begin{pmatrix} \nabla & 0 \\ 0 & \nabla \\ \phi\partial_x & \phi\partial_y \end{pmatrix}$
(M2)	$\frac{\lambda^2}{\lambda^2 + \lVert\nabla f\rVert^2}$	$\frac{1}{2}\int_\Omega (f_t + \nabla f\cdot\mathbf{u})^2$	$\frac{1}{2}\int_\Omega \phi(x,f,\nabla f)(\nabla_H\cdot\mathbf{u})^2 + \sum_{i=1}^{2}\int_\Omega \lvert\nabla u_i\rvert$	$\begin{pmatrix} \nabla & 0 \\ 0 & \nabla \\ \phi\partial_y & -\phi\partial_x \end{pmatrix}$

Using standard dual identities the convex conjugate $F^*(\mathbf{d})$ is computed as

$$F^*(\mathbf{d}) = \frac{1}{2}\|d_3\|_2^2 + \alpha \sum_{i=1}^{2} \delta_{B(L^\infty)}(d_i/\alpha)\,,$$

where $B(L^\infty)$ denotes the unit ball in $L^\infty(\Omega)$ and

$$\delta_{B(L^\infty)}(x^*) = \begin{cases} 0, \text{ if } x^* \in B(L^\infty) \\ +\infty, \text{ otherwise} \end{cases} .$$

Thus the primal-dual formulation is given as

$$\arg\min_{\mathbf{u}} \arg\max_{\mathbf{d}} \ (\mathbf{u}, K^*\mathbf{d}) - \frac{1}{2\beta}\|d_3\|_2^2 - \alpha \sum_{i=1}^{2} \delta_{B(L^\infty)}(d_i/\alpha)\,.$$

Accordingly, the Chambolle-Pock algorithm for this primal-dual problem is given as:

$$\begin{aligned}
\tilde{\mathbf{d}}^{k+1} &= \mathbf{d}^k + \sigma K \bar{\mathbf{u}}\,, \\
\mathbf{d}_{1,2}^{k+1} &= \arg\min_{\mathbf{d}} \left\{ \frac{1}{2}\|\mathbf{d} - \tilde{\mathbf{d}}_{1,2}^{k+1}\|_2^2 + \alpha\sigma\delta_{B(L^\infty)}(\mathbf{d}/\alpha) \right\}, \\
d_3^{k+1} &= \arg\min_{d} \left\{ \frac{1}{2}\|d - \tilde{d}_3^{k+1}\|_2^2 + \frac{\sigma}{2\beta}\|d\|_2^2 \right\}, \\
\tilde{\mathbf{u}}^{k+1} &= \mathbf{u}^k - \tau K^* \mathbf{d}^{k+1}\,, \\
\bar{\mathbf{u}}_{k+1} &= 2\mathbf{u}_{k+1} - \mathbf{u}_k\,.
\end{aligned}$$

To derive the optimality condition for the dual variables d_3, consider the functional

$$J(d_3) = \frac{1}{2}\int_\Omega (d_3 - \tilde{d}_3)^2 + \frac{\sigma}{2\beta}\int_\Omega d_3^2\,.$$

Therefore, setting $d_\theta J = 0$ we get

$$d_3 - \tilde{d}_3 + \frac{\sigma}{\beta} d_3 = 0\,.$$

Rearranging we get

$$d_3^{k+1} = \frac{\beta}{\beta + \sigma}\tilde{d}_3^{k+1}\,.$$

The solution for the indicator function δ is given by the point-wise projections of $\tilde{\mathbf{d}}^{k+1}$, $\mathbf{proj}_\alpha(\tilde{\mathbf{d}}^{k+1})$ onto the unit ball, see [5, 11]. Thus, the iterative scheme for the Chambolle-Pock is given as

$$\tilde{\mathbf{d}}^{k+1} = \mathbf{d}^k + \sigma K \bar{\mathbf{u}}\,,$$

$$\begin{aligned}
\mathbf{d}_{1,2}^{k+1} &= \mathbf{proj}_{\alpha}\left(\tilde{\mathbf{d}}_{1,2}^{k+1}\right), \\
d_3^{k+1} &= \frac{\beta}{\beta+\sigma}\tilde{d}_3^{k+1}, \\
\tilde{\mathbf{u}}^{k+1} &= \mathbf{u}^k - \tau K^*\mathbf{d}^{k+1}, \\
\bar{\mathbf{u}}_{k+1} &= 2\mathbf{u}_{k+1} - \mathbf{u}_k .
\end{aligned}$$

4.2. Optimality Condition for Our Model (M2)

In this case we have

$$G(\mathbf{u}) = \frac{1}{2}\int_{\Omega}(f_t + \nabla f \cdot \mathbf{u})^2, \quad F(K\mathbf{u}) = \frac{1}{2}\int_{\Omega}\phi(f,\nabla f)(\nabla_H \cdot \mathbf{u})^2 + \sum_{i=1}^{2}\int_{\Omega}|\nabla u_i| .$$

The Operator K is given as

$$K\mathbf{u} = \begin{bmatrix} \nabla & 0 \\ 0 & \nabla \\ \phi\partial_y & -\phi\partial_x \end{bmatrix}\begin{bmatrix} u_1 \\ \\ u_2 \end{bmatrix}.$$

Therefore,

$$K^*\mathbf{d} = -\begin{bmatrix} \nabla\cdot & 0 & \partial_y(\phi\cdot) \\ 0 & \nabla\cdot & -\partial_x(\phi\cdot) \end{bmatrix}\begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix}.$$

As before, the convex conjugate $F^*(\mathbf{d})$ is computed as

$$F^*(\mathbf{d}) = \frac{1}{2}\|d_3\|_2^2 + \alpha\sum_{i=1}^{2}\delta_{B(L^\infty)}(d_i/\alpha),$$

Thus the primal-dual formulation is given as

$$\arg\min_{\mathbf{u}}\arg\max_{\mathbf{d}} \ \frac{1}{2}\int_{\Omega}(f_t + \nabla f\cdot\mathbf{u})^2 + (\mathbf{u}, K^*\mathbf{d}) - \frac{1}{2\beta}\|d_3\|_2^2 - \alpha\sum_{i=1}^{2}\delta_{B(L^\infty)}(d_i/\alpha).$$

Accordingly, the Chambolle-Pock algorithm for this primal-dual problem is given as:

$$\begin{aligned}
\tilde{\mathbf{d}}^{k+1} &= \mathbf{d}^k + \sigma K\bar{\mathbf{u}}, \\
\mathbf{d}_{1,2}^{k+1} &= \arg\min_{\mathbf{d}}\left\{\frac{1}{2}\|\mathbf{d} - \tilde{\mathbf{d}}_{1,2}^{k+1}\|_2^2 + \alpha\sigma\delta_{B(L^\infty)}(\mathbf{d}/\alpha)\right\},
\end{aligned}$$

$$
\begin{aligned}
d_3^{k+1} &= \arg\min_d \left\{ \frac{1}{2}\|d - \tilde{d}_3^{k+1}\|_2^2 + \frac{\sigma}{2\beta}\|d\|_2^2 \right\}, \\
\tilde{\mathbf{u}}^{k+1} &= \mathbf{u}^k - \tau K^* \mathbf{d}^{k+1}, \\
\mathbf{u}^{k+1} &= \arg\min_{\mathbf{u}} \left\{ \frac{1}{2}\|\mathbf{u} - \tilde{\mathbf{u}}^{k+1}\|_2^2 + \frac{\tau}{2}\int_\Omega (f_t + \nabla f \cdot \mathbf{u})^2 \right\}, \\
\bar{\mathbf{u}}_{k+1} &= 2\mathbf{u}_{k+1} - \mathbf{u}_k .
\end{aligned}
$$

The optimality conditions for the dual variables follow directly from above. For the primal variable $\mathbf{u}$ the optimality condition can be obtained directly by a quadratic minimization, see [11] for more details. The equations are given as

$$
\begin{aligned}
(1 + \tau f_x^2)u_1 + \tau f_x f_y u_2 &= \tilde{u}_1^{k+1} - \tau f_x f_t , \\
\tau f_x f_y u_1 + (1 + \tau f_y^2)u_2 &= \tilde{u}_2^{k+1} - \tau f_y f_t .
\end{aligned}
$$

Thus the iterative scheme for the Chambolle-Pock is given as

$$
\begin{aligned}
\tilde{\mathbf{d}}^{k+1} &= \mathbf{d}^k + \sigma K \bar{\mathbf{u}}, \\
\mathbf{d}_{1,2}^{k+1} &= \mathbf{proj}_\alpha \left(\tilde{\mathbf{d}}_{1,2}^{k+1} \right), \\
d_3^{k+1} &= \frac{\beta}{\beta + \sigma} \tilde{d}_3^{k+1}, \\
\tilde{\mathbf{u}}^{k+1} &= \mathbf{u}^k - \tau K^* \mathbf{d}^{k+1}, \\
\mathbf{u}^{k+1} &= \left(\frac{b_1 c_3 - c_2 b_2}{c_1 c_3 - c_2^2}, \frac{b_2 c_1 - c_2 b_1}{c_1 c_3 - c_2^2} \right), \\
\bar{\mathbf{u}}_{k+1} &= 2\mathbf{u}_{k+1} - \mathbf{u}_k ,
\end{aligned}
$$

where c_1, c_2, c_3 are the elements of the coefficient matrix given by $c_1 = 1 + \tau f_x^2, c_2 = \tau f_x f_y, c_3 = 1 + \tau f_y^2$, b_1, b_2 are the right hand side values given by $b_1 = \tilde{u}_1^{k+1} - \tau f_x f_t, b_2 = \tilde{u}_2^{k+1} - \tau f_y f_t$. In the next section we will look at the numerical discretization and other implementation details.

5. Results

Having obtained the Chambolle-Pock algorithm for solving the saddle-point problem, we now look at the implementation details. Algorithm 5.1 shows the Chambolle-Pock algorithm for our nonlinear constraint-based refinement model.

Algorithm 5.1

1: Initialize $\tau, \sigma \leftarrow 1/\sqrt{8}, 1/\sqrt{8}$
2: Initialize $\mathbf{u}^0 \leftarrow \mathrm{HS}(f_1, f_2)$, $\mathbf{d}^0 \leftarrow 0$
3: Initialize matrix K
4: **repeat**
5: $\quad \mathbf{u}_{\text{old}} \leftarrow \mathbf{u}$
6: $\quad$ Update matrix K
7: $\quad \tilde{\mathbf{d}} \leftarrow \mathbf{d} + \sigma K \bar{\mathbf{u}}$
8: $\quad d_{1,2} \leftarrow \mathbf{proj}_{\sigma/\alpha}(\tilde{d}_{1,2})$
9: $\quad d_3 \leftarrow \frac{\beta}{\beta+\sigma}\tilde{d}_3$
10: $\quad$ Compute matrix K^*
11: $\quad \tilde{\mathbf{u}} \leftarrow \mathbf{u} - \tau K^* \mathbf{d}$
12: $\quad \mathbf{u} \leftarrow \tilde{\mathbf{u}}$
13: $\quad \bar{\mathbf{u}} \leftarrow 2\mathbf{u} - \mathbf{u}_{\text{old}}$
14: **until** convergence

As mentioned previously this model works in two phases wherein the first phase we obtain a crude-pixel correspondence and subsequently refine this estimate in the next phase driven by additional constraints. The initial Horn and Schunck flow was computed using the Chambolle-Pock algorithm, see [7, 11]. Here we observed that using a forward difference scheme for both spatial and temporal image derivatives f_x, f_y and f_t respectively does not yield a stable discretization. Instead, a forward difference scheme for f_t and a central difference scheme for f_x, f_y does yield a stable numerical scheme. In the next step, the operator matrix K is constructed for updating the dual variables d_1, d_2, d_3 shown in steps 9 and 10. This requires solving two sub-problems, one for d_1, d_2 and the other for d_3.

Now $\nabla u_i = (u_{i_x}, u_{i_y}), i = 1, 2$. The associated dual variable is $d_i = (d_{i,1}, d_{i,2})$. The primal formulation comprises of the total-variation regularization. Accordingly,

$$|\nabla u_i|_{L^1} = |u_{i_x}| + |u_{i_y}| \,.$$

Thus the associated dual norm for the variable d_i gives

$$\|d_i\|_{L^\infty} = \max\{|d_{i,1}|, |d_{i,2}|\} \,.$$

The solution for this minimization is the point-wise projection onto the unit ball corresponding to the dual norm. As shown previously, the convex conjugate of the total variation term is the indicator function $\delta_{L^\infty}(d_i/\alpha)$. The associated convex set is defined by

$$\{d_i : \|d_i/\alpha\| \leq 1\} = \{d_i : \|d_i\| \leq \alpha\}, i = 1, 2 \,.$$

Thus the dual update for $d_{1,2}$ can be obtained by the point-wise projection of $\tilde{d}_{1,2}$ onto $[-\alpha, \alpha]$ (see [11]) as

$$d_{1,2} = \mathbf{proj}_{\sigma/\alpha}(\tilde{d}_{1,2}) = \min(\alpha, \max(-\alpha, \tilde{d}_{1,2}))\,.$$

The sub-problem for d_3 is a linear quadratic minimization problem as discussed in the previous section. In the next step the adjoint operator K^* is constructed to update the primal variable $\mathbf{u}$. The subsequent over-relaxation step $\bar{\mathbf{u}} \leftarrow 2\mathbf{u} - \mathbf{u}_{\text{old}}$ is a particular case for $\theta = 1$ in [7] for easier estimates of the convergence. The algorithm is further simplified if the regularization term is linear. In this case the only difference that occurs is the updation of the dual variable $d_{1,2}$ leading to the optimality condition

$$d_{1,2} = \frac{\alpha}{\alpha + \sigma}\tilde{d}_{1,2}\,.$$

The stopping criterion is determined by computing the normalized error from the primal and dual residues. This error metric was introduced by the authors in [13] and is numerically less expensive. Let $\mathbf{u}^{(k)}, \mathbf{d}^{(k)}$ be the primal and the dual updates after k iterations respectively. Then the primal and dual residues at the k^{th} iteration are computed by the formula:

$$p_{\text{res}}^{(k)} := \left| \frac{\mathbf{u}^{(k)} - \mathbf{u}^{(k+1)}}{\tau} - K^*(\mathbf{d}^{(k)} - \mathbf{d}^{(k+1)}) \right|,$$

$$d_{\text{res}}^{(k)} := \left| \frac{\mathbf{d}^{(k)} - \mathbf{d}^{(k+1)}}{\sigma} - K(\mathbf{u}^{(k)} - \mathbf{u}^{(k+1)}) \right|.$$

Therefore, the normalized error at k^{th} step is obtained as:

$$e^{(k)} = \frac{p_{\text{res}}^{(k)} + d_{\text{res}}^{(k)}}{\mu(\Omega)}\,,$$

where $\mu(\Omega)$ refers to the measure of the domain Ω. Chambolle and Pock [7] also showed that the convergence criterion is fulfilled when $\tau\sigma\|K\|^2 < 1, \theta = 1$. Thus τ and σ need to be chosen accordingly. An optimal numerical upper-bound was obtained by Chambolle [6] which satisfies the above criterion. Accordingly we set $\tau = \sigma = 1/\sqrt{8}$.

The Chambolle-Pock algorithm for the angular accuracy model follows in a similar manner. The main difference lies in the primal update step 12 because of the explicit presence of the data term in the functional. The primal variable $\mathbf{u}$ is updated by solving a quadratic minimization problem discussed previously leading to the following update step,

$$\mathbf{u}^{k+1} = \left(\frac{b_1 c_3 - c_2 b_2}{c_1 c_3 - c_2^2}, \frac{b_2 c_1 - c_2 b_1}{c_1 c_3 - c_2^2} \right),$$

where $c_1 = 1 + \tau f_x^2, c_2 = \tau f_x f_y, c_3 = 1 + \tau f_y^2$, $b_1 = \tilde{u}_1^{k+1} - \tau f_x f_t, b_2 = \tilde{u}_2^{k+1} - \tau f_y f_t$. We now show the results for obtained by implementing the algorithm. The first sequence is the Oseen vortex pair. For more details on the sequence we refer to [18].

Figure 2 shows the velocity magnitude plot obtained for the Oseen vortex pair. The algorithm produces dense flow fields while correctly estimating the vortex cores. We now empirically demonstrate the rate of convergence of both models.

Table 3 shows the number of iterations required by the algorithm to reach the error threshold of ϵ. By an order $O(1/N)$ convergence we mean that the number of iterations required to reach a tolerance ϵ is $O(1/\epsilon)$. This is validated from the above table. For Model (M1), for $\epsilon = 0.1$, the number of iterations required to reach the threshold of 0.1 is a multiple of 10. For Model (M2) it requires a multiple of 10^2 iterations. The table also

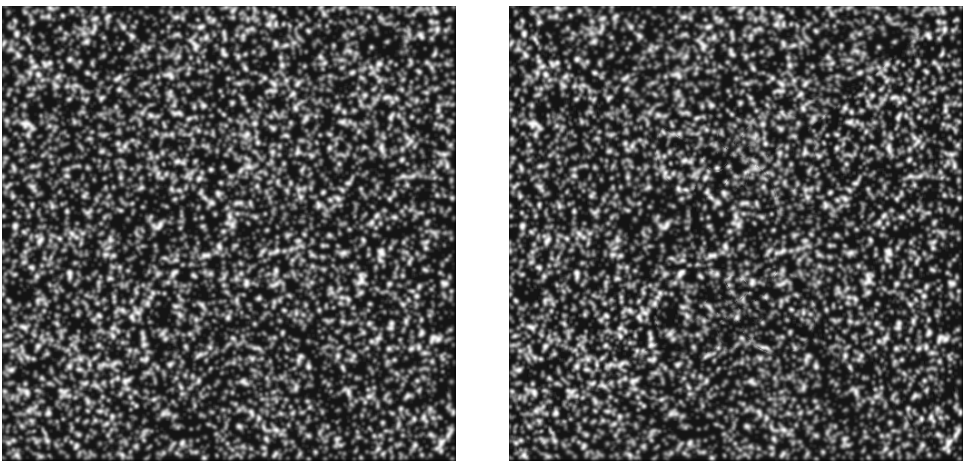


Fig. 1. Oseen vortex pair [18].

Tab. 3. Total numbers of iterations required by the algorithms to reach the threshold of ϵ.

	$\epsilon = 0.1$		$\epsilon = 0.01$	
	Model (M1)	Model (M2)	Model (M1)	Model (M2)
Oseen vortex pair	78	627	755	53753
Cloud sequence	20	99	450	16068
Sphere sequence	10	316	561	18009
Hydrangea	103	346	937	12574
Rubberwhale	42	351	617	13590

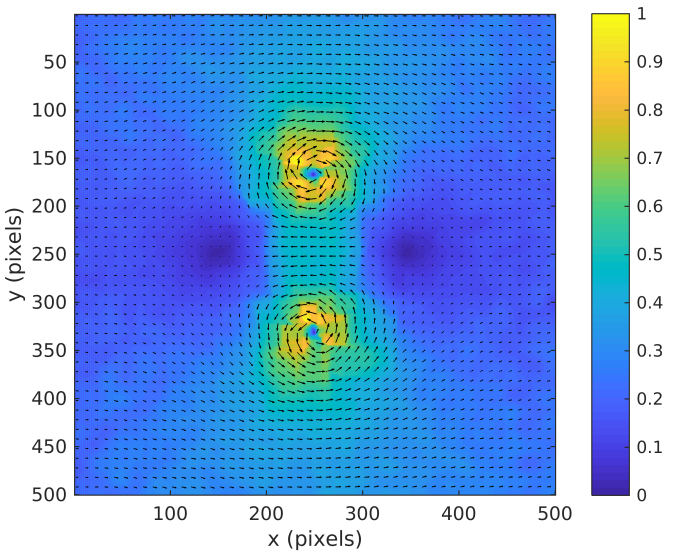


Fig. 2. Velocity plot for the Oseen vortex pair with $\alpha = 0.1, \beta = 0.01, \text{iter} = 50$.

shows that roughly it requires a multiple of 100 iterations for the Model (M1) algorithm to reach the threshold of $\epsilon = 0.01$ and a multiple of 100^2 iterations for Model (M2). The reason for this efficiency can be explained from the fact that in phase 1, the Horn and Schunck initialization brings the solution within a close error range. As a result in the second phase we observe a $O(1/N)$ convergence as mentioned in [7].

5.1. Modern Implementation Principles

Recent developments in optical flow computation reveal that the flow estimates can be significantly improved by incorporating certain established implementation principles. To accommodate these principles in our framework, Algorithm 5.1 is suitably modified to make it a part of a larger implementation procedure.

The computation of image derivatives follows a weighted averaging principle [4, 24]. The current flow estimates are used to warp the second image towards the first using bi-cubic interpolation. The time derivative is the difference between the first image and the warped image. The spatial derivatives are obtained as a weighted average of the first image and the warped image. The weight coefficient is called the blending ratio chosen between 0 and 1.

To account for large displacements of pixel motions, a coarse-to-fine pyramidal scheme is employed [4, 11, 25]. The flow field is first computed at the coarsest level. This

Fig. 3. Estimated flow field for the *rubberwhale* sequence with $\alpha = 10, \beta = 1$.

estimate is upsampled to the next finer level via interpolation and is used to warp the second image towards the first image. The flow increments at this finer level are then computed between the first image and the warped image. This process continues till the finest resolution level is reached. At each pyramid level, 10 warping steps are performed. after each warping iteration, a 5×5 median filter is applied on the flow estimates to remove outliers. Figure 3 shows the obtained flow field for nonlinear refinement using Algorithm 5.1 for the *rubberwhale* sequence.

An improvement is seen in the average angular error (AAE) and the end-point error (EPE) for the nonlinear refinement compared to the linear refinement for the *rubberwhale* sequence as shown in Table 4. The improvement however is not significant which indicates that incorporating the modern implementation principles like coarse-to-fine warping, median filtering and so on also improves the accuracy of the linear refinement process. There are however image sequences for which edge-information are not

Tab. 4. Comparison of the Average Angular Error (AAE) and End Point Error (EPE) for *rubberwhale* sequence († refers to the algorithm implemented with the modern principles).

	Model 1†		Model 2†	
	AAE	**EPE**	**AAE**	**EPE**
Linear Refinement	3.412	0.105	3.410	0.105
Nonlinear Refinement	3.397	0.104	3.355	0.103

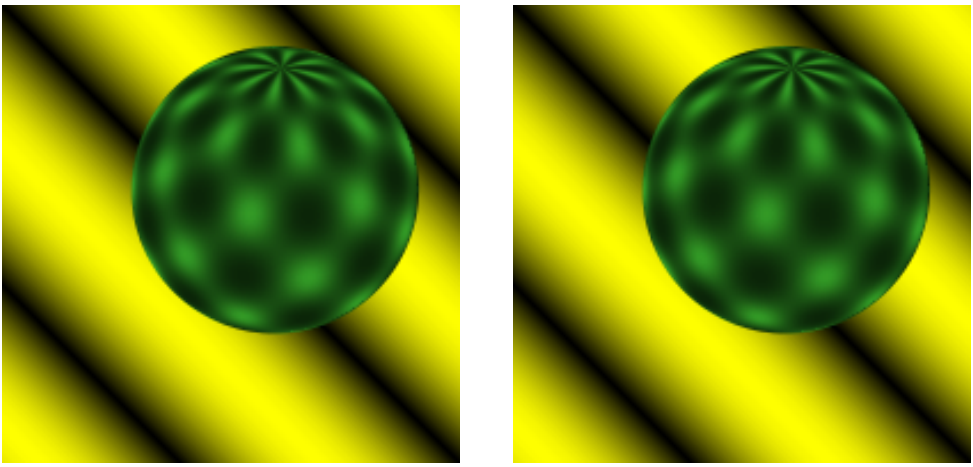
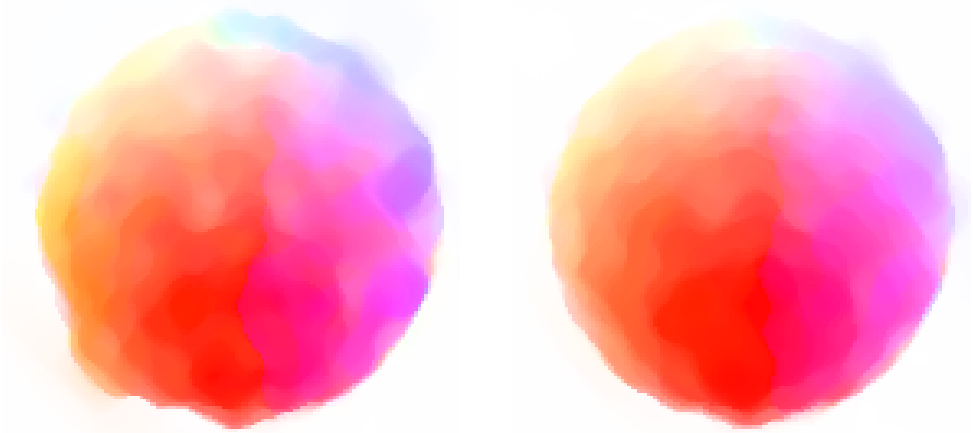


Fig. 4. *Sphere* sequence [22].



(a) Linear Refinement (b) Nonlinear Refinement

Fig. 5. Estimated flow fields of the *sphere* sequence from [22] using Algorithm $1^{\dagger}$ with $\alpha = 1, \beta = 0.1$ († refers to the algorithm implemented with the modern principles).

well-preserved by the linear refinement process. We demonstrate this with the *sphere* sequence [22] (Fig. 4).

Figure 5 shows the color-coded flow estimate for the *sphere* sequence using the Middlebury color coding [2, 3]. The isotropic behaviour is seen in the linear case because of which the edges are not well preserved.

6. Conclusion

In this paper we have proposed two nonlinear variational models for obtaining an accurate estimation of physics-based flow fields such as rotational and fluid flow. The first model is a novel two-phase refinement process where in the first phase a crude estimate is obtained and subsequently refined using additional constraints in the second phase. We have studied the well-posedness of this model using an Evolutionary PDE approach. The second model performs the same refinement using a single functional. We used the first-order primal-dual Chambolle-Pock algorithm for the numerical implementation of the above models. We further empirically demonstrated that the two-phase model leads to a faster convergence rate of the order $O(1/N)$ compared to the second model which has a convergence rate of the order $O(1/N^2)$.

References

[1] G. Aubert, R. Deriche, and P. Kornprobst. Computing optical flow via variational techniques. *SIAM Journal of Applied Mathematics*, 60(1):156–182, 1999. doi:10.1137/S0036139998340170.

[2] S. Baker, D. Scharstein, J. P. Lewis, et al. Optical Flow, 2021. `https://vision.middlebury.edu/flow/`. [Last accessed Dec 2021].

[3] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. Black, and R. Szeliski. A database and evaluation methodology for optical flow. *International Journal of Computer Vision*, 92:1–31, 2011. doi:10.1007/s11263-010-0390-2.

[4] A. Bruhn and J. Weickert. Towards ultimate motion estimation: Combining highest accuracy with real-time performance. In *Proc. 10th IEEE Int. Conf. Computer Vision ICCV 2005*, volume 1, pages 749–755. IEEE, Beijing, China, 17-21 Oct, 2005. doi:10.1109/ICCV.2005.240.

[5] M. Burger, H. Dirks, and L. Frerking. On optical flow models for variational motion estimation. In M. Bergounioux, G. Peyré, C. Schnörr, J. Caillau, and T. Haberkorn, editors, *Variational Methods In Imaging and Geometric Control*, pages 225–251. De Gruyter, Berlin, Boston, 2017. doi:10.1515/9783110430394-007.

[6] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20:89–97, 2004. doi:10.1023/B:JMIV.0000011325.36760.1e.

[7] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications in imaging. *Journal of Mathematical Imaging and Vision*, 40:120–145, 2011. doi:10.1007/s10851-010-0251-1.

[8] I. Cohen. Nonlinear variational method for optical flow computation. In *Proc. 8th Scandinavian Conference on Image Analysis SCIA 1993*, pages 523–530. IAPR, Tromso, Norway, 1993. `https://hal.inria.fr/inria-00615717`.

[9] T. Corpetti, D. Heitz, G. Arroyo, E. Mémin, and A. Santa-Cruz. Fluid experimental flow estimation based on an optical flow scheme. *Experiments in Fluids*, 40:80–97, 2006. doi:10.1007/s00348-005-0048-y.

[10] T. Corpetti, E. Mémin, and P. Pérez. Estimating fluid optical flow. In *Proc. 15th Int. Conf. Pattern Recognition, ICPR 2000*, volume 3, pages 1033–1036. IEEE, Barcelona, Spain, 3-7 Sep, 2000. doi:10.1109/ICPR.2000.903722.

[11] H. Dirks. *Variational Methods for Joint Motion Estimation and Image Reconstruction*. PhD thesis, Wilhems-Universität, 2015.

[12] H. Doshi and N. Uday Kiran. A framework for fluid motion estimation using a constraint-based refinement approach, 2022. arXiv:2011.12267v2. doi:10.48550/arXiv.2011.12267.

[13] T. Goldstein, M. Li, X. Yuan, E. Esser, and R. Baraniuk. Adaptive primal-dual hybrid gradient methods for saddle-point problems, 2015. arXiv:1305.0546v2. doi:10.48550/arXiv.1305.0546.

[14] D. Heitz, E. Mémin, and C. Schnörr. Variational fluid flow measurements from image sequences: Synopsis and perspectives. *Experiments in Fluids*, 48:369–393, 2010. doi:10.1007/s00348-009-0778-3.

[15] W. Hinterberger, O. Scherzer, C. Schnörr, and J. Weickert. Analysis of optical flow models in the framework of calculus of variations. *Numerical Functional Analysis and Optimization*, 23(1-2):69–89, 2002. doi:10.1081/NFA-120004011.

[16] B. K. P. Horn and B. G. Schunck. Determining optical flow. *Artificial Intelligence*, 17(1-3):185–203, 1981. doi:10.1016/0004-3702(81)90024-2.

[17] A. Kumar, A. Tannenbaum, and G. Balas. Optical flow: A curve evolution approach. *IEEE Transactions of Image Processing*, 5(4):598–610, 1996. doi:10.1109/83.491336.

[18] T. Liu. OpenOpticalFlow: An open source program for extraction of velocity fields from flow visualization images. *Journal of Open Research Software*, 5(1):29, 2017. doi:10.5334/jors.168.

[19] T. Liu and L. Shen. Fluid flow and optical flow. *Journal of Fluid Mechanics*, 614:253–291, 2008. doi:10.1017/S0022112008003273.

[20] A. Luttman, E. M. Bollt, R. Basnayake, S. Kramer, and N. B. Tufillaro. A framework for estimating potential fluid flow from digital imagery. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 23(3):033134, 2013. doi:10.1063/1.4821188.

[21] A. Martin, E. Schiavi, and S. Segura de León. On 1-Laplacian elliptic equations modeling magnetic resonance imaging Rician denoising. *Journal of Mathematical Imaging and Vision*, 57:202–224, 2017. doi:10.1007/s10851-016-0675-3.

[22] B. McCaine. Optical Flow Algorithm Evaluation. Computer Vision Research Group, Department of Computer Science, University of Otago Dunedin, New Zealand, 2021. `http://of-eval.sourceforge.net/`. [Last accessed Dec 2021].

[23] C. Schnörr. Determining optical flow for irregular domains by minimizing quadratic functionals of a certain class. *International Journal of Computer Vision*, 6:25–38, 1991. doi:10.1007/BF00127124.

[24] D. Sun, R. Roth, and M. J. Black. Secrets of optical flow estimation and their principles. In *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition CVPR 2010*, pages 2432–2439. IEEE, San Francisco, CA, USA, 13-18 Jun, 2010. doi:10.1109/CVPR.2010.5539939.

[25] A. Wedel, T. Pock, C. Zach, H. Bischof, and D. Cremers. An improved algorithm for TV-L^1 optical flow. In D. Cremers et al., editors, *Statistical and Geometrical Approaches to Visual Motion Analysis. Proc. International Dagstuhl Seminar*, volume 5604 of *Lecture Notes in Computer Science*, pages 23–45, Dagstuhl Castle, Germany, Jul 13-18, 2008. Springer Berlin Heidelberg, 2009. doi:10.1007/978-3-642-03061-1_2.

[26] L. Wei, R. P. Agarwal, and P. J. Y. Wong. Existence and iterative construction of solutions to nonlinear Dirichlet boundary value problems with p-Laplacian operator. *Complex Variables and Elliptic Equations: An International Journal*, 55(5-6):601–608, 2010. doi:10.1080/17476930802657632.

[27] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime TV-L^1 optical flow. In

F. A. Hamprecht et al., editors, *Pattern Recognition. Proc. 29th DAGM Joint Pattern Recognition Symposium*, volume 4713 of *Lecture Notes in Computer Science*, pages 214–223, Heidelberg, Germany, 12-14 Sep, 2007. Springer Berlin Heidelberg. doi:10.1007/978-3-540-74936-3_22.

[28] B. Zhang and Z. Zhu. A primal-dual algorithm framework for convex saddle-point optimization. *Journal of Inequalities and Applications*, Article number: 267, 2017. doi:10.1186/s13660-017-1548-z.

[29] X. Zhang, M. Burger, and S. Osher. A unified primal-dual algorithm framework based on Bregman iteration. *Journal of Scientific Computing*, 46:20–46, 2011. doi:10.1007/s10915-010-9408-8.

[30] J. Zhao, Y. Wang, and H. Wang. Optical flow with harmonic constraint and oriented smoothness. In *Proc. 6th Int. Conf. Image and Graphics ICIG 2011*, pages 94–99, Hefei, China, 12-15 Aug, 2011. doi:10.1109/ICIG.2011.122.