# MolGrapher: Graph-based Visual Recognition of Chemical Structures

Lucas Morin[1, 2] Martin Danelljan[2] Maria Isabel Agea[1] Ahmed Nassar[1]
Valery Weber[1] Ingmar Meijer[1] Peter Staar[1] Fisher Yu[2]
[1]IBM Research [2]ETH Zurich
{lum, ahn, vwe, inm, taa}@zurich.ibm.com martin.danelljan@vision.ee.ethz.ch i@yf.io

## Abstract

*The automatic analysis of chemical literature has immense potential to accelerate the discovery of new materials and drugs. Much of the critical information in patent documents and scientific articles is contained in figures, depicting the molecule structures. However, automatically parsing the exact chemical structure is a formidable challenge, due to the amount of detailed information, the diversity of drawing styles, and the need for training data. In this work, we introduce MolGrapher to recognize chemical structures visually. First, a deep keypoint detector detects the atoms. Second, we treat all candidate atoms and bonds as nodes and put them in a graph. This construct allows a natural graph representation of the molecule. Last, we classify atom and bond nodes in the graph with a Graph Neural Network. To address the lack of real training data, we propose a synthetic data generation pipeline producing diverse and realistic results. In addition, we introduce a large-scale benchmark of annotated real molecule images, USPTO-30K, to spur research on this critical topic. Extensive experiments on five datasets show that our approach significantly outperforms classical and learning-based methods in most settings. Code, models, and datasets are available* [1].

## 1. Introduction

The creation of an open-source database offering a unified view of our current knowledge of all studied molecules, would greatly accelerate research and development in numerous fields, ranging from the pharmaceutical industry to semiconductor manufacturing. Currently, information about a molecule's properties is distributed across various databases, research articles, and chemical patents, each measuring different subsets of properties. In such documents, molecules are most often described using images, by drawing their molecular structures. Automatic parsing of molecules from document images, known as Optical Chemical Structure Recognition (OCSR), is thus a critical task in the quest towards establishing a large-scale digital molecule database.

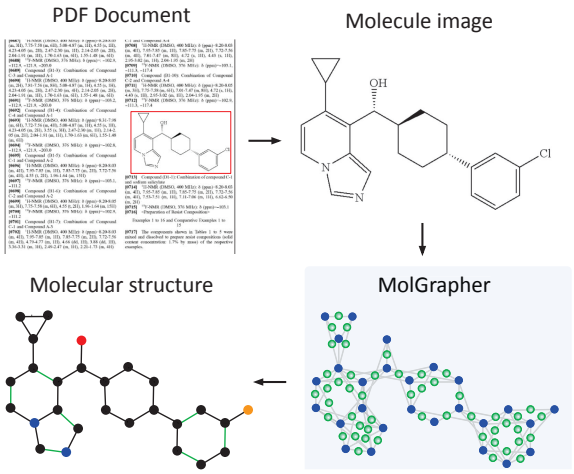


Figure 1. **MolGrapher** extracts the chemical structure, including all atoms and bonds, from a molecule image in a document. Our approach constructs a supergraph of the molecule (bottom right), containing all detected atom and bond candidates. These nodes are then classified by a Graph Neural Network in order to retrieve the chemical structure.

OCSR, however, poses multiple key challenges, which limit the accuracy of current solutions. First, a molecule can be drawn with a variety of styles and conventions. Even the projection of the molecular structure from 3-D onto a 2-D drawing is not unique, leading to the development of various projection algorithms with diverse results. Secondly, OCSR requires the extraction of detailed information from the image. Molecules often contain even hundreds of atoms and bonds, which all need to be correctly classified and linked. Thirdly, molecules have a virtually endless diversity stemming from the combinatorical explosion of possibilities, presenting particular challenges to data-driven deep learning methods. Lastly, available real training datasets are extremely limited, further complicating the application of deep learning.

[1]https://github.com/DS4SD/MolGrapher

Initial OCSR approaches followed the graph reconstruction paradigm. These methods start by extracting the fundamental components of molecules, such as atoms, bonds, and charges, often using hand-crafted image processing algorithms [8, 29, 21]. Then, rules are applied to connect these elements to reconstruct a graph representation of the molecule. Some recent works [19, 33, 38] introduce deep learning elements into the atom and bond detection steps. However, these approaches still rely on hand-crafted rules to link the recognized constituents, and thus not fully benefit from the advantages brought by deep learning.

Most recently, approaches instead follow the image captioning [4, 24, 34] paradigm by applying a deep network to directly output a character string identifying the molecule, e.g. in the form of SMILES [32]. However, these methods do not explicitly exploit the rich priors and invariances provided by the graph structure. The SMILES representation does not encode the geometrical and neighborhood structure of the molecule graph in a natural manner. While a recent work [36] predicts a graph in an autoregressive manner, adding one atom at a time prevents from fully exploiting the graph structure. Partially due to the lack of such a strong inductive bias, image captioning based methods still lag behind from the performance of state-of-the-art rule-based methods [8, 29, 21] on standard benchmarks [8, 9, 28, 22]. Moreover, the performance of captioning based methods severely degrade when increasing the size and complexity of the molecule, as they struggle to recover the detailed information in the image.

We introduce MolGrapher for Optical Chemical Structure Recognition, illustrated in Fig. 1. Our approach explicitly utilizes the graph structure as a strong inductive bias, while also allowing for the final molecule structure itself to be predicted by a deep learning model. MolGrapher operates in three steps. First, we locate atoms and abbreviations in the image with a keypoint detector network. Given the estimated locations of the atoms, we form a supergraph of the molecular structure, where two types of nodes represent candidate atoms and bonds respectively (see Fig. 1). Unlike previous approaches, we do not commit to a final molecule structure at this point, instead extracting a larger set of candidate bonds, forming a superset of the final molecular graph. Candidates that do not correspond to any bond are removed in the final prediction step by including a 'no bond' class.

In the final step, we input the constructed supergraph into a Graph Neural Network (GNN). We first extract deep node embeddings through a backbone network. The GNN then operates on the neighborhood structure provided by the supergraph in order to integrate visual information with learned chemistry priors. We then read-out the results with MLPs, classifying each node into a set of atom and bond classes. We also identify abbreviations through a separate class, which are parsed by an OCR component. Since training data is scarce, we develop, and release, a synthetic training data generation pipeline, capable of generating diverse and challenging examples to ensure robust generalization to a wide variety of drawing styles.

To further aid the research in the community, we introduce USPTO-30K, a large-scale benchmark dataset of annotated real molecule images. It contains separate subsets, in order to independently study the recognition of simple molecules, abbreviated molecules and extremely large molecules. We perform comprehensive experiments on five benchmarks: our USPTO-30K, USPTO [8], Maybridge UoB [28], CLEF-2012 [22] and JPO [9]. We outperform all deep learning based methods that only utilizes synthetic training. Our approach even surpass previous methods that rely on finetuning on real data on most benchmarks, and outperform the longstanding state-of-the-art by rule-based methods [8, 29, 21] in most settings.

## 2. Related work

OCSR approaches can be divided in two main categories.
**Image captioning based OCSR.** Most of the recent end-to-end deep learning approaches are based on image captioning. These models use an encoder to extract visual features from the image and a decoder to translate them to a SMILES [32] or InChI [10] sequence. More precisely, DECIMER [26], MICER [35], MSE-DUDL [30] and Img2Mol [4] are using a convolutional encoder and a recurrent decoder which is respectively, a GRU decoder, a LSTM decoder, a GridLSTM decoder, and a RNN decoder. Later works proposed replacing the recurrent decoder with a transformer encoder-decoder, including DECIMER 1.0 [27], SwinOCSR [34], IMG2SMI [3] and Image2SMILES [13]. Other image captioning methods are solely based on transformers, using a vision transformer in [31], a Swin transformer in [23] or a Deep TNT transformer encoder with a transformer decoder in ICMDT [17].

One drawback of image captioning methods is the need of large training sets. The SMILES representation has numerous downsides for a learning task [15], one of them which is ambiguity due to the association of very different string identifiers to similar molecules. By predicting SMILES, instead of a more faithful representation of the image, the model needs to learn jointly to recognize the image and understand the SMILES language. Multiple string identifiers, possibly suited for a learning task have been proposed, including SELFIES, DeepSMILES [25] or abstract mathematical identifiers [4]. Nevertheless, representing a molecule image as a string adds extra complexity for a learning task. Additionally, these models to not properly handle uncertainty by predicting incorrect, but still valid, molecules for challenging inputs such as large molecules.

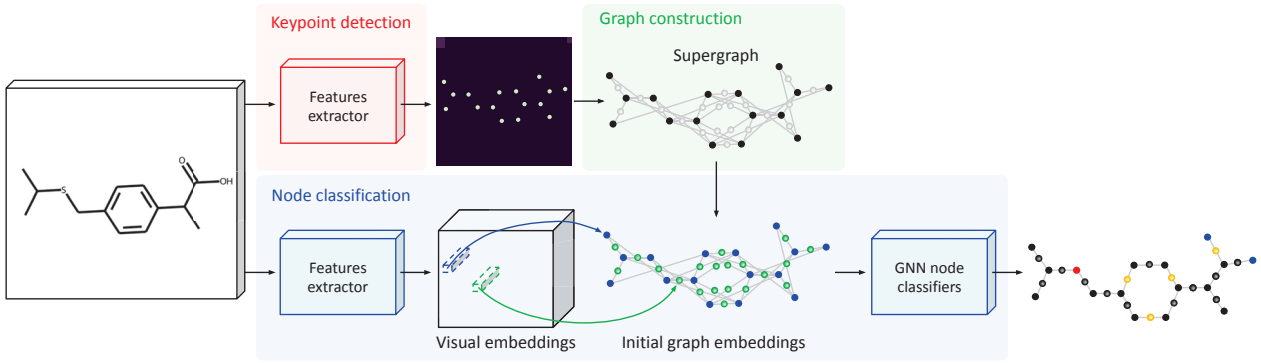


Figure 2. **Molecular graph recognition architecture.** We illustrate the architecture of MolGrapher, a graph-based network for Optical Chemical Structure Recognition. The keypoint detector (red) locates atoms nodes in the molecule. A supergraph containing atom and bond candidates is constructed (green). Atoms and bonds are classified using a Graph Neural Network (blue).

Yet, to parse scientific literature at large scale, avoiding these false positives is critical. This behaviour is explained by the fact that the model is trained to always output valid SMILES, while using global image features. Contrary to these approaches, our model extracts precise visual features for each elementary component of a molecule and operate on a graph, allowing substantially better recognition performance for large molecules, possibly larger than the ones used for training.

**Graph reconstruction based OCSR.** Traditional approaches for Optical Chemical Structure Recognition rely on hand-crafted image processing algorithms to detect elementary components of the molecules and connect them to reconstruct the molecular graph [2, 8, 12, 20, 28, 29, 21]. More recent works detect elementary components using machine learning models. For instance, ChemGrapher [19] or ABC-Net [38] use convolutional segmentation networks while MolMiner [33] employs a YOLO object detector. As studied in [11], detection based methods can be trained with fewer training samples. By doing multiple low-level detections, it is possible to supervise precisely the training. It also offers a natural way to inject prior knowledge, or impose chemistry rules to the model. Our method benefits from these advantages but differentiates itself by learning both the detection and the association of the fundamental components of molecules.

Image2Graph [36] uses a transformer to build the graph in an auto-regressive way, by adding one atom at the time. [23] predicts a SMILES sequence, but enriched with atoms positions, and connectivity information, drifting away from the string identifier and getting closer to a graphical representation. Our new graph reconstruction approach do not build a graph auto-repressively. We first detects all possible components and associations, and then classify them. It allows to use localized visual features, as well as a complete neighborhood contextual information for each atom and bond prediction.

## 3. MolGrapher

We introduce MolGrapher, a graph-based network for Optical Chemical Structure Recognition. Our model architecture is illustrated in Figure 2. Our pipeline consists of three steps. Firstly, the keypoint detector locates atoms nodes in the molecule. Secondly, we construct a graph containing atom and bond candidates. Finally, atoms and bonds are classified using a Graph Neural Network.

### 3.1. Keypoint Detection

In order to construct the molecular graph, we first need to localize keypoints in the image. Keypoints refer to the position of atoms and 'superatoms', i.e. abbreviated groups of atoms. In the later stages, this enables the extraction of relevant visual features for atoms and bonds, as well as the construction of our molecule supergraph.

Our keypoint detector predicts a heatmap, where each peak corresponds to an atom location. The output of a feature extractor is passed through convolutional layers to predict the final single-channel heatmap of the atom locations. To extract the peak locations, we first threshold the heatmap by removing any value in the bottom $10^{\text{th}}$ percentile. We then collect regional maxima, using a window size of $5 \times 5$, as the final atom locations.

Atoms are labeled with Gaussian functions, allowing smoother supervision as well as capturing a degree of uncertainty regarding the predicted positions. Since the ground-truth heatmaps predominantly contains background values, we use the weight-adaptive heatmap regression loss [18] to overcome the class imbalance. It reduces the impact of well-classified samples, which mostly constitute background, allowing the training to focus on harder samples.

### 3.2. Supergraph Construction

In this section, we describe the construction of the supergraph, which contains all atom and bond candidates to be classified. The supergraph is composed of two types of

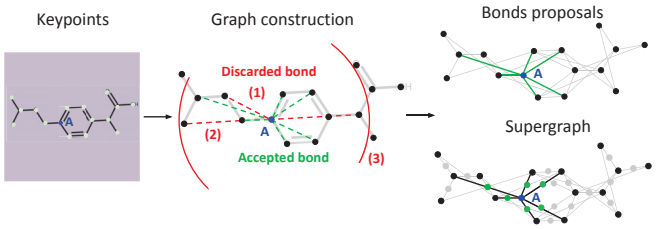


Figure 3. **Supergraph construction.** The figure presents the construction of bonds proposals for an atom denoted $A$. Considered bonds are depicted with dashed lines. Green bonds are accepted in the supergraph, while red bonds are discarded because: (1) there are no filled pixels around their centerpoints or (2) they are obstructed by other keypoints.

nodes: atoms and bonds. Further, each edge in the graph only connects an atom node to a bond node. Representing both of these entities as nodes allows us to process them uniformly through standard message passing layers.

The input to our supergraph construction is a set of approximate atom locations. Since the connectivity between the detected atoms is unknown, our graph has to represent a superset of the final molecule prediction, thus termed 'supergraph'. That is, it contains all possible bonds. By including a prediction class corresponding to 'no bond', bond candidates which are not actually in the molecule can be removed in the later stage. Including an analogous class for 'no atom', our approach can even correct false positives predicted by the keypoint detection network.

Our supergraph construction is illustrated in Figure 3. We first consider each keypoint individually, adding connections to all other keypoints within a radius of three times the estimated bond length [8]. Subsequently, we remove connections which have no filled pixels around their centerpoints or if there is a third keypoint located between its two extremities. Finally, we keep at most six bond candidates for each atom, by eliminating the longest connections. Chemistry prior knowledge ensures that it is extremely unlikely for an atom to have more than five neighboring bonds. Our settings are highly conservative to avoid deleting any actual bond. On the other hand, pruning superfluous bonds avoids any unnecessary false positives in the later stage and a more effective connectivity for the GNN.

### 3.3. Node Classification

In order to classify atom and bond nodes, we integrate a Graph Neural Network (GNN) that operates on the constructed supergraph. The embeddings for all nodes are initialized using visual features as well as their types. This approach leverages the local context of atoms and bonds to predict their class. Intuitively, it enables the learning of chemistry rules, i.e. atoms have a fixed valence, as well as chemistry prior knowledge, i.e. some typical substructures are more common than others.

Let $e_i^1 \in R^D$ denotes the initial embedding of the node $i$ of type $t_i \in \{\text{atom}, \text{bond}\}$ located at position $(x_i, y_i)$. It is computed by combining a visual feature vector $v_i$ and a learnable type encoding $w_{t_i}$,

$$e_i^0 = v_i + w_{t_i} \tag{1}$$

The visual features $v_i$ are extracted from a deep feature map $F$, extracted by a backbone network. For an atom node $i$ we evaluate the feature map in the corresponding location $(x_i, y_i)$ with bilinear interpolation,

$$v_i = F(x_i, y_i)\,. \tag{2}$$

For a bond node $i$, we aggregate features at locations along the bond as

$$v_i = \frac{1}{J}\sum_{j=1}^{J} F(x_i^j, y_i^j)\,, \tag{3}$$

where $(x_i^j, y_i^j)$ are the locations of $J$ uniformly spaced points along the bond itself.

Given initial node embeddings, information is propagated through the graph to learn local dependencies between atoms and bonds. Indeed, directly neighboring atoms and bonds respect strict chemistry rules, and larger neighborhoods form patterns called functional groups, which are typically abundant in nature. We employ GNN layers, denoted $\{g^k\}_{k\in[1,N]}$, to iteratively update the node embeddings,

$$e^{k+1} = g^k(e^k)\,. \tag{4}$$

The final predictions $\{p_i\}$ are then obtained using two Multi-Layer Perceptrons, one for each node type $t$,

$$p_i = MLP_t(e_i^N)\,. \tag{5}$$

The vector $p_i \in \mathbb{R}^{C_t}$ contains the logits for the final atom or bond classes.

The most represented bond types include 'no bond', 'single', 'double' and 'triple'. Atom classes include the common (e.g. O) and exotic atoms as well as charged atoms (e.g. $O^-$). Since molecule images often include whole groups in the nodes, e.g. COOH, we further include a class 'superatom' to recognize these cases. Such nodes are then parsed as described in Section 3.4.

Our node classification model is trained using the cross-entropy loss for both the atom and bond prediction heads. During training, the supergraph structure used as input is obtained by applying the graph construction algorithm on augmented ground truth keypoints. To simulate possible errors from the keypoint detector, we add noise to the atom positions and randomly include false-positive keypoints. Decoupling the keypoint detection and the node classification allow to train faster and to use optimized training sets for each task.

### 3.4. Superatom Groups Recognition

To recognize abbreviated substructures, named superatoms, we use an external Optical Character Recognition (OCR) system. Relying on an external OCR system provides strong robustness, since the model is trained on a diversity of text representations that cannot necessarily be generated in a molecule image. At the location of an abbreviated group, the node classifier predicts a 'superatom' class. Then, the OCR engine PP-OCR [6] is used to recognize the text written at this location, which is finally replaced by its corresponding sub-molecule, stored in a precomputed mapping.

## 4. Datasets

### 4.1. Synthetic Training Data Generation

Training datasets of annotated molecule images extracted from scientific publications are extremely limited. Therefore, we develop a synthetic training pipeline to generate a broad diversity of molecular representations. It further allows us to extract atom-level annotations.

Our dataset is created using molecule SMILES retrieved from the database PubChem [14]. To increase the probability that the dataset covers the largest variety of molecular structures, we sample SMILES using a strategy based on functional groups. A functional group is molecule substructure typically abundant in nature. Thus, we retrieve molecules from PubChem containing each functional group in a defined collection of 1540 functional groups. To further increase diversity, these molecules are queried with different ranges of number of atoms. Finally the dataset is filtered by removing salts, molecules containing complex polycyles, isotopes or radical electrons. Training images are then generated from SMILES using the molecule drawing library RDKit [16].

To capture the large diversity of drawing styles and conventions from different scientific documents, the synthetic set is highly augmented, at multiple levels. Firstly, molecules are randomly transformed, notably by setting the selection of a molecular conformation, adding artificial superatom groups with single or multiple attachment points or displaying solid, dashed or wavy bonds. Secondly, the rendering parameters used in RDKit are randomly set, including the font, the bond width, the display of aromatic cycles using circles. This leads to substantial variations in the drawn molecule, to aid generalization to real datasets.

The generated molecule images are saved together with their corresponding MolFiles. A MolFile [5] stores information about the atoms and their positions in a molecule, as well as bonds and their connectivity. It provides the necessary information for supervising the training. This generation pipeline covers a large diversity of molecules including stereo-chemistry and molecules with superatom groups. Finally, the images undergo several image augmentations on the fly such as gaussian blurring, the adding of pepper patches, random lines, and random captions.

The exhaustive lists of molecule, rendering and image augmentations, as well as details regarding the generation of atom-level annotations are available in the supplementary materials.

### 4.2. The USPTO-30K Dataset

Existing benchmarks have some limitations. Being created using only a few documents, they contain batches of very similar molecules. For example in a patent, a molecule could typically be displayed together with all the substituent of one particular substructure, resulting in large batches of almost identical molecules. Additionally, the existing sets contain molecules of different kinds, including superatom groups and various markush [7] features, which should be evaluated independently. In practice, it is important to delimit on which types of molecules models can be applied.

We introduce USPTO-30K, a large-scale benchmark dataset of annotated molecule images, which overcomes these limitations. It is created using the pairs of images and MolFiles [5] by the United States Patent and Trademark Office (USPTO) [1]. Each molecule was independently selected among all the available images from 2001 to 2020. The set consists of three subsets to decouple the study of clean molecules, molecules with abbreviations and large molecules. USPTO-10K contains 10,000 clean molecules, i.e. without any abbreviated groups. USPTO-10K-abb contains 10,000 molecules with superatom groups. USPTO-10K-L contains 10,000 clean molecules with more than 70 atoms. We provide visualizations and analysis in the supplementary materials.

## 5. Experiments

In this section, we perform comprehensive experiments on multiple OCSR benchmarks. Moreover, we propose an analysis of the different components of our method.

### 5.1. Implementation details

The implementation is done with PyTorch 1.12 with CUDA 11.3. For the keypoint detector features extractor, we use a ResNet-18 backbone with $8\times$ dilatation factor [37] to preserve a high spatial resolution. For the node classifier features extractor, we resort to a ResNet-50 with a $2\times$ dilation factor. The model is trained on 300,000 synthetic images, presented in subsection 4.1. We train for 20 epochs on 3 NVIDIA A100 GPUs using ADAM with a learning rate of 0.0001 that we decay after 5000 iterations by a factor of 0.8. The losses for atoms and bonds classifiers are weighted by factors 1 and 3, respectively. During inference, we pre-process images by removing captions using PP-OCR [6]. Additionally, in case an initial prediction is an

Table 1. **Comparison of our method with existing OCSR models.** We report the accuracy, i.e. the percentage of perfectly recognized molecule images, on datasets coming from real scientific documents. ∗: re-implemented results. †: results from original publications. ‡: results from [11].

| Method | USPTO (5719) | Maybridge UoB (5740) | CLEF-2012 (992) | JPO (450) | USPTO-10K (10 000) | USPTO-10K-Abb (10 000) | USPTO-10K-L (10 000) |
|---|---|---|---|---|---|---|---|
| *Rule-based methods* | | | | | | | |
| OSRA 2.1 [8] * | 89.3 | 86.3 | **93.4** | 56.3 | 89.7 | 63.9 | 43.1 |
| MolVec 0.9.7 [21] * | 89.1 | 88.3 | 81.2 | 66.8 | 92.4 | 70.3 | **64.0** |
| Imago 2.0 [29] * | 89.4 | 63.9 | 68.2 | 41.0 | 89.9 | 63.0 | 47.3 |
| *Only synthetic training* | | | | | | | |
| DECIMER 2.0 [24] † | 61.0 | 88.0 | 72.0 | 64.0 | - | - | - |
| Image2Graph [36] † | 44.9 | 72.0 | 37.8 | 24.0 | - | - | - |
| Graph Generation [23] † | 67.0 | 83.1 | 74.6 | - | - | - | - |
| CEDe [11] ‡ | 79.0 | 74.1 | 68.0 | 49.4 | - | - | - |
| ChemGrapher [19] ‡ | 80.9 | 83.2 | 75.5 | 53.3 | - | - | - |
| Img2Mol [4] * | 25.2 | 68.0 | 17.9 | 16.1 | 35.4 | 13.8 | 0.0 |
| **MolGrapher (Ours)** | **91.5** | **94.9** | 90.5 | **67.5** | **93.3** | **82.8** | 31.4 |

invalid molecule, we use PP-OCR to merge keypoints located in a same detected text cell. See the supplementary material for more details.

## 5.2. Evaluation datasets and metrics

To compare our method with state-of-the-art, the model is evaluated on the standard benchmarks USPTO [8], Maybridge UoB [28], CLEF-2012 [22] and JPO [9]. USPTO and CLEF are collections of 5,719 and 992 molecule images. Besides, JPO contains 450 images published by the Japanese Patent Office (JPO). JPO is particularly challenging because of its non-standard drawing conventions and poor images qualities. Lastly, Maybridge UoB is a dataset of 5,740 scanned molecule images taken from a catalogue of drug compounds by Maybridge. To compare methods, we compute accuracy, defined as the percentage of perfectly recognized molecules. In practice, this is done by verifying that the predicted and ground-truth molecules have identical InChI [10] keys. For evaluation, stereo-chemistry is removed and markush structures are not considered.

## 5.3. State-of-the-art Comparison

Table 1 compares the OCSR methods on different benchmarks. Our method achieves superior results compared to rule-based models on most datasets, including USPTO, Maybridge UoB and JPO. Solely using synthetic training, our method reduces the error rate by more than half compared to other deep learning methods on USPTO, Maybridge UoB, and CLEF-2012. The design of our model confers a strong generalization ability. In practice, this is very important as our model will be applied to a wide variety of documents, which are not necessarily evaluated by the current benchmark datasets. MolGrapher also maintains good performance for extremely large molecules in USPTO-10K-L, which only contains examples with more than 70 atoms. On the contrary, recognizing large molecules is a significant challenge for image captioning based methods. For instance, Clevert et al. point out that the performance of Img2Mol [4] drop sharply for molecules larger than 40 atoms. This is confirmed in our evaluation, where it fails to correctly recognize virtually any molecule in USPTO-10K-L. Table 2 also provides comparison with models finetuned using real data. Our model surpasses these methods on Maybridge UoB, CLEF-2012 and JPO and is second best on USPTO, while not relying on any real data for training.

Table 2. **Comparison of our method with deep OCSR models finetuned on real data.** We report the accuracy, i.e. the percentage of perfectly recognized molecule images, on datasets coming from real scientific documents. †: results from original publications. ‡: results (in grey) from unavailable sub-splits of the original benchmarks, only reported for reference.

| Method | USPTO (5719) | Maybridge UoB (5740) | CLEF-2012 (992) | JPO (450) |
|---|---|---|---|---|
| *Real data finetuning* | | | | |
| Image2Graph [36] † | 55.1 | 83.0 | 51.7 | 50.0 |
| Graph Generation [23] † | **92.9** | 86.6 | 87.5 | - |
| CEDe [11] † ‡ | 91.0 | 91.5 | 86.6 | 74.0 |
| *Only synthetic training* | | | | |
| **MolGrapher (Ours)** | 91.5 | **94.9** | **90.5** | **67.5** |

Our model outperforms existing methods on our USPTO-10K and USPTO-10K-Abb. The performance is particularly improved for molecules containing abbreviated groups. The performance gap between USPTO and USPTO-10K-Abb for rule-based approaches suggests that these methods are specifically parameterized for abbreviations in the existing benchmark datasets. Our model is the only deep learning method to provide a competitive performance for extremely large molecules. This is allowed by our architecture, which extracts localized visual features.

## 5.4. Model Robustness

Clevert *et al.* [4] introduced modified version of the standard benchmarks by applying slight perturbations, such as rotation and shearing, to the input images. Table 3 presents a comparison to previous methods on the same perturbed sets. The augmented images can be seen as a simulation

Table 3. **Evaluation of model robustness.** We report the accuracy, i.e. the percentage of perfectly recognized molecule images, on perturbed datasets coming from real scientific documents. †: results from [4].

| Method | USPTO$^{p}$ (4852) | Maybridge UoB$^{p}$ (5716) | CLEF-2012$^{p}$ (711) | JPO$^{p}$ (365) |
|---|---|---|---|---|
| OSRA 2.1 [8] † | 6.4 | 70.9 | 17.0 | 33.0 |
| MolVec 0.9.7 [21] † | 30.7 | 75.0 | 44.5 | 49.5 |
| Imago 2.0 [29] † | 5.1 | 5.1 | 26.7 | 23.2 |
| Img2Mol [4] † | 42.3 | 78.2 | 48.8 | 45.1 |
| **MolGrapher (Ours)** | **86.7** | **94.1** | **87.8** | **55.4** |

of scanned images, which are typically found in chemical patents. We observe that the performance of rule based approaches, OSRA, MolVec and Imago, decreases sharply. However, our model maintains a good result, outperforming other methods by a significant margin. This robustness is essential for large-scale analysis of chemical literature, as input molecule image quality cannot be controlled.

## 5.5. Qualitative evaluation

In this section, we conduct a qualitative evaluation of MolGrapher, in comparison to state-of-the-art methods. First, Figure 4 illustrates the intermediate outputs of MolGrapher. Note that MolGrapher accurately recognizes molecules in challenging cases with overlapping bonds, without the use of any post-processing rules. Figure 5 showcases examples of predicted molecules for images from various benchmark datasets. MolGrapher demonstrates robustness to images containing captions (Figure 5 row 1). It can correctly recognize extremely large molecules (Figure 5 row 4), and is robust to solid, dashed or wavy bonds (Figure 5 row 5). Unlike image captioning based methods such as as Img2Mol, our predictions preserve the projection and atom placement used in the input image (Figure 5 row 2). This can convey critical information for human interpretation.

## 5.6. Ablation study

We conduct an ablation study to demonstrate the impact of each component in the proposed method.

**Synthetic training set.** We evaluate the impact of the three levels of augmentation of the training set, illustrated in Table 4. The image augmentations have a major impact on the performance of the model. In practice, we noticed that the keypoint detector benefits greatly from using heavy image transformations. Additionally, the molecule augmentations, which allow training molecules to contain abbreviated groups, proved crucial to handle images containing abbreviated groups, such as in USPTO.

Table 4. **Training set analysis.** Impact of training set augmentations on performance. Molecule, rendering and image level augmentation are independently removed.

| | USPTO | JPO | JPO$^{p}$ |
|---|---|---|---|
| All augmentations | **91.5** | **67.5** | **55.4** |
| No molecule augmentation | 52.4 | 52.7 | 43.4 |
| No rendering augmentation | 78.4 | 55.7 | 44.1 |
| No image augmentation | 69.8 | 29.3 | 22.5 |

**Keypoint detector.** In Table 5, we experiment with different standard deviations for the Gaussian keypoint labels, and keypoint regression losses. Using a large standard deviation of $b/5$, where $b$ denotes the bond length, creates overlaps between keypoints and significantly decreases performance. While the model is not sensitive to the choice of training loss, the WAHR loss offers a slight advantage.

**Supergraph construction.** In Table 7, we evaluate the impact of the maximum number of bond candidates and the search radius in the supergraph construction. We observe that our approach is not sensitive to these settings, but only experiences a small drop when increasing the number of bond candidates. This drop is explained by false positives predicted during the node classification stage.

**Node classifier.** As demonstrated in Table 7, initializing nodes embeddings with both visual and type encoding increases performance. We also experimented with positional

Table 5. **Keypoint detector analysis.** We analyse the standard deviation of the Gaussian functions (top), as well as the training loss function (bottom). $b$ denotes the molecule bond length.

| | | USPTO | JPO | JPO$^{p}$ |
|---|---|---|---|---|
| Heatmap standard deviation | $b/5$ | 81.3 | 54.7 | 46.8 |
| | $b/10$ | **91.5** | **67.5** | **55.4** |
| | $b/15$ | 91.4 | 64.7 | 51.6 |
| Training loss | L2 | 91.1 | 66.2 | 55.1 |
| | WAHR [18] | **91.5** | **67.5** | **55.4** |

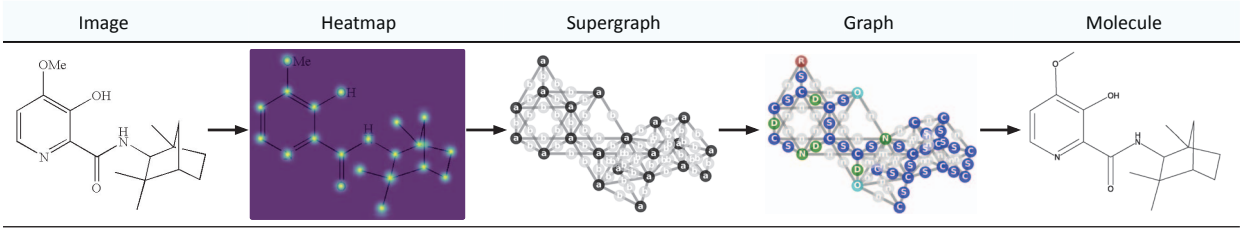


Figure 4. **Prediction steps.** Keypoints are detected and then used to build a supergraph. After classifying the nodes of the graph and recognizing abbreviated groups, the output molecule is created. In this example, the polycyclic molecule contains overlapping bonds, a challenging feature for OCSR models, and is still correctly recognized.

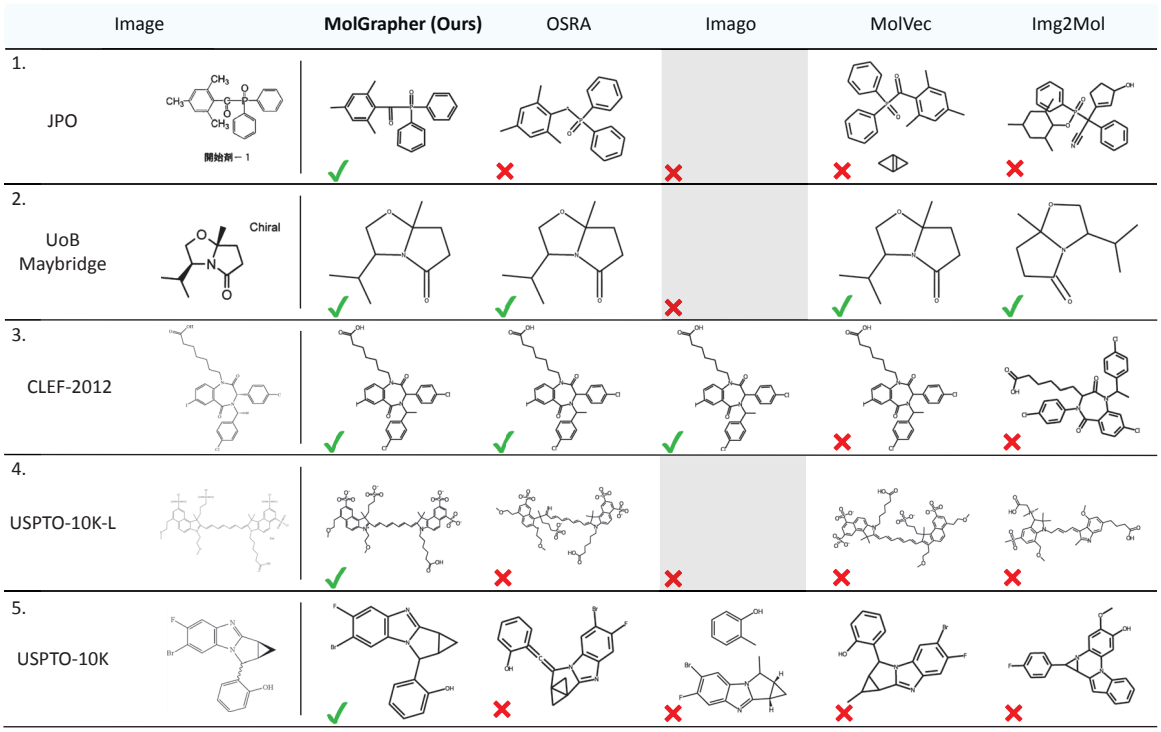


Figure 5. **Qualitative comparison.** The figure shows examples of predictions for characteristic images from different benchmarks. Compared to previous rule-based and learning based methods, our approach robustly recognizes the exact molecular structure in challenging cases, such as with distracting captions, stereo-chemistry, and very large molecules.

Table 6. **Supergraph construction analysis.** We analyse the maximum number of bond candidates (top), and the maximum search radius (bottom). $b$ denotes the molecule bond length.

| | | USPTO | JPO | JPO$^{\text{P}}$ |
|---|---|---|---|---|
| Number bond candidates | 6 | **91.5** | **67.5** | **55.4** |
| | 10 | 91.0 | 65.7 | 54.1 |
| Search radius | $3b$ | **91.5** | **67.5** | **55.4** |
| | $5b$ | 91.3 | 65.0 | 53.1 |

Table 7. **Node classifier analysis.** We analyse the features representing graph nodes (top), and the usage of GCN layers (bottom). †: without abbreviated molecules and charges.

| | | USPTO | JPO | JPO$^{\text{P}}$ |
|---|---|---|---|---|
| Node embedding | Visual | 90.0 | 64.7 | 54.9 |
| | + Type | **91.5** | **67.5** | **55.4** |
| | + Position | 91.3 | 62.8 | 54.8 |
| | | USPTO$^{\dagger}$ | JPO$^{\dagger}$ | JPO$^{\text{P}\dagger}$ |
| Number of GCN Layers | 0 | 90.1 | 58.0 | 53.6 |
| | 4 | **90.4** | **66.6** | **61.6** |

embedding but found it to reduce the overall performance. The model demonstrating, in average, best performances does not include GCN layers. However, as demonstrated in Table 7, for a limited evaluation setup, GCN layers have positive impact on performance, leading to significant gains of 8.6% and 8.0% on JPO and JPO$^{\text{P}}$ when using 4 instead of 0 GCN layers. Note that even the version using 0 GNN layers, still employs our supergraph for feature aggregation

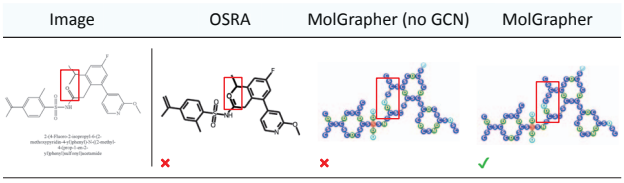


Figure 6. **Ablation experiment.** The figure shows MolGrapher predictions with and without using GCN layers, illustrating that GCN layers allows the model to learn chemistry rules. Indeed, the model correctly connect the oxygen atom highlighted in red to a double bond only.

and classification. The propagation of information through the graph is beneficial for challenging cases, such as the one illustrated in Figure 6. In this example, interpreting only the visual information would be misleading, even for humans. Understanding chemistry rules is mandatory to resolve the ambiguities in the drawing, which is correctly done by our model.

## 6. Conclusion

We propose a novel architecture for recognizing 2-D molecule depictions in documents by exploiting the natural graph representation of molecules and graph symmetries. Our model accurately detects atoms and bonds based on their visual features and local context, outperforming exist-

ing methods on standard benchmarks and our new USPTO-30K dataset. The model is trained on synthetic images and demonstrates strong generalization capabilities, making it useful at scale without requiring annotations for each journal or patent office.

## References


[1] United states patent and trademark office. http://uspto.gov. Accessed: 1 January 2023.

[2] Syed Saqib Bukhari, Zaryab Iftikhar, and Andreas Dengel. Chemical structure recognition (csr) system: Automatic analysis of 2d chemical structures in document images. In 2019 International Conference on Document Analysis and Recognition (ICDAR), pages 1262–1267. IEEE, 9 2019.

[3] Daniel Campos and Heng Ji. IMG2SMI: translating molecular structure images to simplified molecular-input line-entry system. CoRR, abs/2109.04202, 2021.

[4] Djork-Arné Clevert, Tuan Le, Robin Winter, and Floriane Montanari. Img2mol – accurate smiles recognition from molecular graphical depictions. Chem. Sci., 12:14174–14181, 9 2021.

[5] Arthur Dalby, James G. Nourse, W. Douglas Hounshell, Ann K. I. Gushurst, David L. Grier, Burton A. Leland, and John Laufer. Description of several chemical structure file formats used by computer programs developed at molecular design limited. Journal of Chemical Information and Computer Sciences, 32(3):244–255, May 1992.

[6] Yuning Du, Chenxia Li, Ruoyu Guo, Xiaoting Yin, Weiwei Liu, Jun Zhou, Yifan Bai, Zilin Yu, Yehua Yang, Qingqing Dang, and Haoshuang Wang. PP-OCR: A practical ultra lightweight OCR system. CoRR, abs/2009.09941, 2020.

[7] Tommy. Ebe, Karen A. Sanderson, and Patricia S. Wilson. The chemical abstracts service generic chemical (markush) structure storage and retrieval capability. 2. the marpat file. Journal of Chemical Information and Computer Sciences, 31(1):31–36, Feb 1991.

[8] Igor V Filippov and Marc C Nicklaus. Optical structure recognition software to recover chemical information: Osra, an open source solution. J. Chem. Inf. Model., 49(3):740–743, March 2009.

[9] Akio Fujiyoshi, Koji Nakagawa, and Masakazu Suzuki. Robust method of segmentation and recognition of chemical structure images in cheminfty. Pre-proceedings of the 9th IAPR international workshop on graphics recognition, GREC, 01 2011.

[10] Stephen R. Heller, Alan McNaught, Igor Pletnev, Stephen Stein, and Dmitrii Tchekhovskoi. Inchi, the iupac international chemical identifier. Journal of Cheminformatics, 7(1):23, May 2015.

[11] Rodrigo Hormazabal, Changyoung Park, Soonyoung Lee, Sehui Han, Yeonsik Jo, Jaewan Lee, Ahra Jo, Seung Hwan Kim, Jaegul Choo, Moontae Lee, and Honglak Lee. CEDe: A collection of expert-curated datasets with atom-level entity annotations for optical chemical structure recognition. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track, 2022.

[12] P. Ibison, M. Jacquot, F. Kam, A. G. Neville, R. W. Simpson, C. Tonnelier, T. Venczel, and A. P. Johnson. Chemical literature data extraction: The clide project. Journal of Chemical Information and Computer Sciences, 33(3):338–344, May 1993.

[13] Ivan Khokhlov, Lev Krasnov, Maxim Fedorov, and Sergey Sosnin. Image2smiles: Transformer-based molecular optical recognition engine. Chemistry–Methods, 2, 01 2022.

[14] Sunghwan Kim, Jie Chen, Tiejun Cheng, Asta Gindulyte, Jia He, Siqian He, Qingliang Li, Benjamin A Shoemaker, Paul A Thiessen, Bo Yu, Leonid Zaslavsky, Jian Zhang, and Evan E Bolton. Pubchem 2019 update: improved access to chemical data. Nucleic Acids Res., 47(D1):D1102–D1109, January 2019.

[15] Mario Krenn, Qianxiang Ai, Senja Barthel, Nessa Carson, Angelo Frei, Nathan C. Frey, Pascal Friederich, Théophile Gaudin, Alberto Alexander Gayle, Kevin Maik Jablonka, Rafael F. Lameiro, Dominik Lemm, Alston Lo, Seyed Mohamad Moosavi, José Manuel Nápoles-Duarte, AkshatKumar Nigam, Robert Pollice, Kohulan Rajan, Ulrich Schatzschneider, Philippe Schwaller, Marta Skreta, Berend Smit, Felix Strieth-Kalthoff, Chong Sun, Gary Tom, Guido Falk von Rudorff, Andrew Wang, Andrew D. White, Adamo Young, Rose Yu, and Alán Aspuru-Guzik. Selfies and the future of molecular string representations. Patterns, 3(10):100588, 10 2022.

[16] Greg Landrum. Rdkit: Open-source cheminformatics software. http://www.rdkit.org/. Accessed: 1 January 2023.

[17] Yanchi Li, Guanyu Chen, and Xiang Li. Automated recognition of chemical molecule images based on an improved tnt model. Applied Sciences, 12(2):680, 1 2022.

[18] Zhengxiong Luo, Zhicheng Wang, Yan Huang, Liang Wang, Tieniu Tan, and Erjin Zhou. Rethinking the heatmap regression for bottom-up human pose estimation. In CVPR, 2021.

[19] Martijn Oldenhof, Adam Arany, Yves Moreau, and Jaak Simm. Chemgrapher: Optical graph recognition of chemical compounds by deep learning. Journal of Chemical Information and Modeling, 60(10):4506–4517, Oct 2020.

[20] Tom Y. Ouyang and Randall Davis. Chemink: A natural real-time recognition system for chemical drawings. In Proceedings of the 16th International Conference on Intelligent User Interfaces, IUI '11, page 267–276, New York, NY, USA, 2 2011. Association for Computing Machinery.

[21] Tyler Peryea, Daniel Katzel, Tongan Zhao, Noel Southall, , and Dac-Trung Nguyen. Molvec. https://github.com/ncats/molvec, 2013. Accessed: 1 January 2023.

[22] Florina Piroi, Mihai Lupu, A. Hanbury, Walid Magdy, Alan Sexton, and I. Filippov. Clef-ip 2012: Retrieval experiments in the intellectual property domain. CEUR Workshop Proceedings, 1178, 01 2012.

[23] Yujie Qian, Zhengkai Tu, Jiang Guo, Connor W. Coley, and Regina Barzilay. Robust molecular image recognition: A graph generation approach. CoRR, abs/2205.14311, 2022.

[24] Kohulan Rajan, Henning Otto Brinkhaus, M Isabel Agea, Achim Zielesny, and Christoph Steinbeck. Decimer.ai - an open platform for automated optical chemical structure identification, segmentation and recognition in scientific publications. feb 2023.

[25] Kohulan Rajan, Christoph Steinbeck, and Achim Zielesny. Performance of chemical structure string representations for chemical image recognition using transformers. Digital Discovery, 1:84–90, 2022.

[26] Kohulan Rajan, Achim Zielesny, and Christoph Steinbeck. Decimer: towards deep learning for chemical image recognition. Journal of Cheminformatics, 12(1):65, Oct 2020.

[27] Kohulan Rajan, Achim Zielesny, and Christoph Steinbeck. Decimer 1.0: deep learning for chemical image recognition using transformers. Journal of Cheminformatics, 13(1):61, Aug 2021.

[28] Noureddin M. Sadawi, Alan P. Sexton, and Volker Sorge. Chemical structure recognition: a rule-based approach. In Christian Viard-Gaudin and Richard Zanibbi, editors, Document Recognition and Retrieval XIX, volume 8297, page 82970E. International Society for Optics and Photonics, SPIE, 1 2012.

[29] Viktor Smolov, Fedor Zentsev, and Mikhail Rybalkin. Imago: Open-source toolkit for 2d chemical structure image recognition. In Ellen M. Voorhees and Lori P. Buckland, editors, Text Retrieval Conference. National Institute of Standards and Technology (NIST), 2011.

[30] Joshua Staker, Kyle Marshall, Robert Abel, and Carolyn M. McQuaw. Molecular structure extraction from documents using deep learning. Journal of Chemical Information and Modeling, 59(3):1017–1029, Mar 2019.

[31] Carola Sundaramoorthy, Lin Ziwen Kelvin, Mahak Sarin, and Shubham Gupta. End-to-end attention-based image captioning. CoRR, abs/2104.14721, 4 2021.

[32] David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. Journal of Chemical Information and Computer Sciences, 28(1):31–36, Feb 1988.

[33] Youjun Xu, Jinchuan Xiao, Chia-Han Chou, Jianhang Zhang, Jintao Zhu, Qiwan Hu, Hemin Li, Ningsheng Han, Bingyu Liu, Shuaipeng Zhang, Jinyu Han, Zhen Zhang, Shuhao Zhang, Weilin Zhang, Luhua Lai, and Jianfeng Pei. Molminer: You only look once for chemical structure recognition. Journal of Chemical Information and Modeling, 62(22):5321–5328, 11 2022. PMID: 36108142.

[34] Zhanpeng Xu, Jianhua Li, Zhaopeng Yang, Shiliang Li, and Honglin Li. Swinocsr: end-to-end optical chemical structure recognition using a swin transformer. Journal of Cheminformatics, 14(1):41, Jul 2022.

[35] Jiacai Yi, Chengkun Wu, Xiaochen Zhang, Xinyi Xiao, Yanlong Qiu, Wentao Zhao, Tingjun Hou, and Dongsheng Cao. Micer: a pre-trained encoder-decoder architecture for molecular image captioning. Bioinformatics, 38(19):4562–4572, September 2022.

[36] Sanghyun Yoo, Ohyun Kwon, and Hoshik Lee. Image-to-graph transformers for chemical structure recognition. In ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 3393–3397, 2022.

[37] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. In Yoshua Bengio and Yann LeCun, editors, 4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings, 2016.

[38] Xiao-Chen Zhang, Jia-Cai Yi, Guo-Ping Yang, Cheng-Kun Wu, Ting-Jun Hou, and Dong-Sheng Cao. Abc-net: a divide-and-conquer based deep learning architecture for smiles recognition from molecular images. Brief. Bioinform., 23(2), March 2022.