

Recognizing Planar Kinematic Mechanisms from a Single Image using Evolutionary Computation

Matthew Eicholtz
Department of Mechanical
Engineering
Carnegie Mellon University
Pittsburgh, PA 15213
meicholt@andrew.cmu.edu

Levent Burak Kara
Department of Mechanical
Engineering
Carnegie Mellon University
Pittsburgh, PA 15213
lkara@cmu.edu

Jason Lohn
Department of Electrical and
Computer Engineering
Carnegie Mellon University -
Silicon Valley
Moffett Field, CA 94035
jlohn@ece.cmu.edu

ABSTRACT

In this paper, a method is presented that automatically recognizes kinematic mechanisms from textbook images using an evolutionary algorithm to complement computer vision techniques for object detection. Specifically, a nondominated sorting genetic algorithm (NSGA-II) is used to optimize the number and position of mechanical joints in an image and corresponding joint connections (*i.e.* rigid bodies) such that Pareto front solutions maximize image consistency and mechanical feasibility. A well-known object detector is used as an example method for locating joints, and local image features between pairwise detected joints are used to predict likely connections. The performance of the algorithm using these specific vision techniques is compared to a parameterized detection scheme in order to decouple the efficacy of the object detector from the evolutionary algorithm. Experiments were performed to validate this approach on selected images from a custom dataset, and the results demonstrate reasonable success in both accuracy and speed.

Categories and Subject Descriptors

G.1.6 [Numerical Analysis]: Optimization—*Constrained optimization, Stochastic programming*; I.4.8 [Image Processing and Computer Vision]: Scene Analysis—*Object Recognition*

Keywords

computer vision; evolutionary multiobjective optimization; kinematic simulation; object recognition

1. INTRODUCTION

Visualizing the motion characteristics of a multibody mechanical system is an important step in design analysis and synthesis [9]. Often, engineers use mental simulations to

infer mechanical behavior [17], but this can be a difficult for complex problems [16] or for individuals with low spatial ability [18]. Alternatively, computer simulations can be generated using specialized software [1, 21], but this task is often time-consuming, which limits its usefulness in certain situations (*e.g.* students solving a dynamics problem, engineers brainstorming potential design concepts), and may require advanced software skills, which hinders novice users.

In the present work, we propose a method for quickly generating simulation models by automatically identifying the underlying mechanical structure in single images. Researchers have briefly studied the automatic recognition of mechanical systems from sketches [8, 13, 14], but these approaches typically involve clean images, well-defined part templates, and sometimes can make use of temporal information to aid recognition. Raw textbook images of kinematic mechanisms, on the other hand, are often noisy, may contain extraneous information from other graphics, and do not always contain well-defined part models.

To our knowledge, the automatic recognition of kinematic mechanisms from textbook images is a novel problem in both computer vision and evolutionary computation, and it has many interesting challenges. One of the primary challenges in tackling this problem is determining the best mechanism representation for recognition. Typical vision-based detection methods treat an object as a single part and learn a model using low-level image features [7, 20, 28], but this technique would not work well for mechanical linkages, which may vastly differ in appearance from one to the next. More recent vision approaches use part-based models [6, 11, 12], which treat the object of interest as a collection of parts to be recognized. While this is more in line with the structure of mechanisms, which can be viewed as a collection of rigid bodies connected by joints that constrain their motion, these methods require learned spatial relationships (deformable models) between the parts. In the present domain, this is impractical given that there are an infinite number of possible mechanism configurations. The problem is compounded by the fact that model parameters of a mechanism, in general, are unknown *a priori*; the number of joints, type of joints, number of bodies, and configuration can all vary.

With this in mind, we propose solving the problem of mechanism identification in two stages. The system takes as input a raw textbook image containing a single, planar, kinematic mechanism of unknown structure. In the first stage,

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Copyright 20XX ACM X-XXXXX-XX-X/XX/XX ...\$15.00.

joints are detected using a conventional object detector, and low-level image features are used to predict the likelihood of joint connections (*i.e.* rigid bodies). In the second stage, we use a multiobjective evolutionary algorithm (MOEA) to optimize the detected parts, while simultaneously evolving structures that are mechanically feasible. The result is a small set of feasible mechanisms that can be easily integrated into engineering software packages for kinematic simulation and analysis.

The remainder of the paper is structured as follows: section 2 highlights related work in object detection and evolutionary computation. Specific algorithm details are provided in section 3. Experiments and concluding remarks are given in sections 4 and 5, respectively.

2. RELATED WORK

Object detection is an important task for computer vision that has been widely studied in recent years. A typical approach is to generate a large number of training images containing the object of interest, extract salient features from those images [2, 3, 7, 20], train a model on those features, and then scan test images using the model to locate the object. One of the primary challenges when designing an object detector is determining which features are most relevant. In this paper, a popular feature descriptor based on oriented gradients is used [7]. This descriptor was originally developed for human detection and utilized in numerous applications since. The basic steps for computing HOG features are as follows: color normalization (*if desired*), compute gradients over entire image, compute weighted histograms over small cells, merge and normalize overlapping blocks of cell histograms (each cell typically contributes to 4 block normalizations), and collect features (*vectorize if needed*). In the original work, HOG features were computed on a set of training images (both with and without humans) and used to train a soft ($C = 0.01$) linear Support Vector Machine (SVM), which attempts to find the most discriminative hyperplane in feature space to separate the two classes (human/no human). The present study implements this method as an initial step to identifying mechanism structure.

There is a growing body of research in the area of multiobjective evolutionary algorithms (MOEAs), especially in regard to real-world applications. Many well-known MOEAs are based on Pareto dominance [10, 27, 30, 29], which states that a given solution (U) dominates another solution (V) if it is at least as good on all objectives and better on at least one objective; that is,

$$U \succeq V \iff \forall i \in \{1, \dots, n\} u_i \geq v_i, \text{ and } \exists i \in \{1, \dots, n\}, u_i > v_i \quad (1)$$

where n is the number of objectives, and (u_i, v_i) is the score of individuals (U, V) on the i th objective. Arguably the most popular MOEA of this type and the one used in this present work is the nondominated sorting genetic algorithm (NSGA-II) [10], which has been successful largely due to fast computation of nondominated fronts, preservation of elitist solutions, and lack of a user-specified sharing parameter.

The nondominated sorting genetic algorithm was first introduced in almost 20 years ago (NSGA, [27]) and improved 8 years later (NSGA-II, [10]). The two identifying characteristics of NSGA-II are nondominated sorting and crowding distance. Nondominated sorting involves locating the Pareto

front (all nondominated solutions), assigning those solutions a rank of 1, and iteratively assigning higher ranks to each Pareto front level, ignoring all previously detected levels. Crowding distance measures the local spread of solutions and is used to preserve diversity in the population. These parameters are used during the selection process as follows. When sorting a set of solutions, any solution with lower rank goes before solutions with higher rank. Inevitably, there will be cases when the desired number of survivors cuts through one of the Pareto ranks. In this case, the solutions with that rank are sorted with preference to higher crowding distances. This algorithm has been shown to have success when the number of objectives is small, so it should be a suitable approach for the problem presented in this paper.

3. TECHNICAL DETAILS

The proposed approach to mechanism identification in images combines vision strategies for object detection with a popular multiobjective evolutionary algorithm for optimization. In this section, specific details are provided for each component of the algorithm.

3.1 Vision-based Detection of Mechanical Components

In this work, the key to identifying the true mechanism contained in an image lies in the ability to accurately detect two classes of objects: joints and rigid bodies. While previous works in parts-based detection might suggest we learn detectors for each class separately [11], the present domain is ill-suited for this approach because rigid bodies exhibit high variance in image features such as size, shape, and color, making the task of learning a reliable rigid body detector difficult. Instead, we make use of the fact that knowing the location of likely joints in an image greatly reduces the space of likely rigid body locations because all rigid bodies must be connected to at least two joints¹. With this in mind, the general pipeline for the vision portion of the algorithm is as follows: (*i*) detect likely locations of mechanical joints; then (*ii*) evaluate the likelihood of a rigid body between each pair of joints.

3.1.1 Detecting mechanical joints

The first step in the vision pipeline is to identify the locations of each joint type² in an input image. We accomplish this task by running a sliding window over the entire image and classifying each patch using a linear SVM. The SVM is trained over HOG features [7] computed on a set of example images.

The training data comprises positive and negative image patches extracted from selected images. The positive examples contain the joint type of interest and are taken from hand-labeled images of mechanisms, while the negative examples are randomly selected from the same images, making sure they do not overlap with positive examples. Similar to previous work on supervised learning algorithms [13], we augment the training set by applying a series of simple

¹One exception to this rule is a pendulum, for which a rigid body has only one joint connection, but this case is excluded from the current domain of interest due to lack of complexity

²For all experiments described in this paper, we limit the domain exclusively to revolute joints, but in theory, incorporating additional joint types (*e.g.* prismatic joints) would simply require training a new detector.

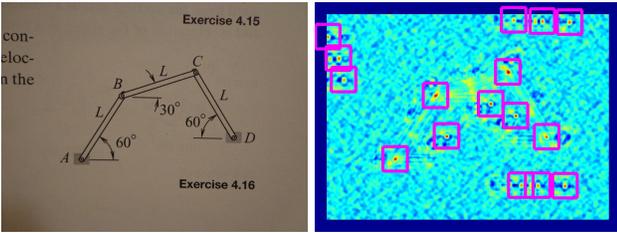


Figure 1: (a) An example image and (b) the resulting heat map of joint detections, where color represents detection strength and rectangles highlight local maxima.

transformations to each positive example: reflection about the vertical or horizontal axis, and rotation by 90, 180, and 270 degrees. This effectively increases the positive training data by a factor of 5, which likely improves the discriminative power of the joint detector. In addition, we follow the approach of Dalal and Triggs [7] to improve classification accuracy by mining hard negatives from an initially trained SVM. This is especially critical for our problem domain because most of the background in textbook images is blank, and therefore the initial set of negative training examples may not accurately reflect the diversity of the negative image space.

After training, the SVM can be used on a test image to classify the image patch centered at each pixel. The result is a heat map, in which distance from the decision boundary (hyperplane) encodes detection strength. In our method, we apply the mean shift procedure [5] to isolate local maxima and then discard extrema with strength less than zero (negative distance to the hyperplane indicates the patch is more likely to *not* contain a joint). Figure 1 illustrates this concept on an example image. The remaining detected joint locations and strengths are then passed to later stages of the algorithm.

3.1.2 Detecting rigid bodies

As mentioned previously, rigid bodies are not explicitly recognized using a sliding window detector; rather, they are identified by the local image features that exist between pairs of detected joints. Specifically, we use a normalized measure of geodesic time to determine the strength of a pairwise joint connection (see Figure 2). The geodesic time [25] between any pair of pixels (p, q) in a grayscale image is defined as

$$t_g(p, q) = \min\{t_g(P) | P \text{ links } p \text{ to } q\} \quad (2)$$

and the geodesic time of a given path P of length n connecting two pixels is given by

$$t_g(P) = \frac{I_{p_0}}{2} + \frac{I_{p_n}}{2} + \sum_{i=1}^{n-1} I_{p_i} \quad (3)$$

where I_{p_j} is the pixel intensity of the j th pixel on the path. Without loss of generality, let us assume we want to maximize geodesic time (*i.e.* true joint connections should have high values). In a typical mechanism, there generally exists a path through darker regions for pairs of joints that are located on the same rigid body (especially along solid boundary lines). With this in mind, we invert the grayscale image so that dark regions in the raw image have higher

values of geodesic time. Then, we normalize the result using the Euclidean distance transform with one of the joints in question acting as the seed location. Without this normalization, the algorithm would strongly bias connections between joints that are far away from each other, even if the connection is false.

One strength of this metric is its ability to identify rigid bodies of varying shape and size. However, due to the nature of kinematic chains, it also tends to misclassify false connections for joints that are on neighboring bodies. To overcome this challenge, we increase the pixel intensity in the raw image of each patch surrounding the detected joints, proportional to SVM strength. This has the effect of decreasing geodesic time for paths between joints that intersect other joints. Also, to accommodate small errors in the location of detected joints, which may undesirably increase geodesic time, we artificially darken a small region (*e.g.* 16-pixel radius) surrounding the joint center to ensure the beginning of all paths emanating from that joint are equally strong.

3.1.3 Parameterized detection scheme

While the aforementioned vision-based detection scheme is likely a good practical solution to the problem of mechanism recognition, it may lead to a misleading evaluation of the evolutionary algorithm, since the two processes are not independent. In order to decouple the performance of the computer vision and evolutionary computation aspects of our approach, we propose the parameterization of detection results as follows.

1. *Precision* (Pr): $Pr = t_p / (t_p + f_p)$
2. *Difference in mean strength* ($\Delta\mu$): $\mu_{fp} - \mu_{tp}$
3. *Standard deviation of true positive strength* (σ_{tp})
4. *Standard deviation of false positive strength* (σ_{fp})

In theory, these parameters could apply to any object detector. For the present work, detection of joints and connections are coupled, such that precision is only defined once, yielding a 7-dimensional parameter space (4 for joints, 3 for rigid bodies). Given any 7-tuple corresponding to these parameters, we can artificially generate strength values for positive and negative examples of joints and connections, without regard to actual image features or the spatial characteristics of joints. In this manner, the efficacy of the evolutionary computation can be evaluated without specifying a specific vision-based algorithm for object detection.

3.2 Evolutionary-based Optimization

Given the output from the object detection pipeline, which is a set of strength values associated with joints and pairwise joint connections, the problem becomes one of constrained multiobjective optimization. Specifically, we seek to find a hypothesis of a mechanism that is strongly consistent with what has been detected in the image as well as reasonable in terms of kinematic simulation. To that end, we employ the nondominated sorting genetic algorithm (NSGA-II) introduced by Deb *et al.* [10]. This algorithm was selected for several reasons, including its well-known success in solving real-world applications, its ability to quickly find a diverse set of good solutions for multiple conflicting objectives, and the unique opportunities it provides for handling constraints. Notable details specific to the present domain are outlined below.

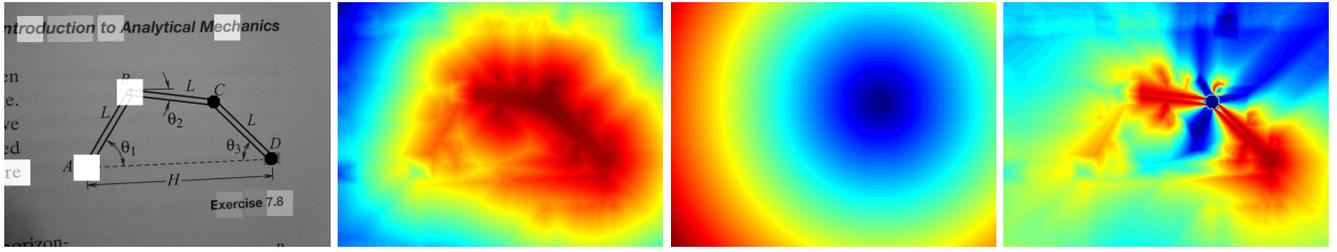


Figure 2: Computation of normalized geodesic time. (a) The sample grayscale image, with the pair of joints involved in the calculation identified by black circles and all other joint patches lightened proportional to their detection strength. (b) Geodesic time transform divided by (c) Euclidean distance transform yields (d) the normalized geodesic time. In all instances, stronger values are more red while weaker values are more blue.

3.2.1 Representation

We represent a mechanism using a graph structure, similar to [19], but with nodes representing joints and edges corresponding to rigid bodies. There are several interesting properties of this representation. In particular, the presence of an edge indicates that the pair of joints associated with the connecting nodes are located on the same rigid body. It is not always true, however, that all edges correspond to unique bodies; in fact, any body containing more than two joints will contribute a fully connected subgraph comprising multiple nodes and edges. In addition, we relax the constraint that multiple pins fixed to the ground must share an edge because a typical image may not exhibit strong visual cues that grounded pins are on the same body. Consequently, the graph representation for a given mechanism is not necessarily unique; also, multiple mechanisms can be represented by the same graph.

Figure 3 illustrates the phenotype-genotype mapping for our problem. The genotype is a vectorized transformation of the mechanism graph, and it consists of two parts: (i) a bit-string of length N to indicate the presence of a joint in an individual hypothesis, where N is the number of detected joints; and (ii) a bit string of length $N(N-1)/2$, which encodes the presence of pairwise edges. The latter part is derived from the upper triangular matrix of the adjacency matrix, which is symmetric. It should be noted that the genotype does not explicitly include any information regarding the spatial layout of detected joints. Also, this representation allows an edge to exist in the absence of one or more of its connecting nodes. To deal with this issue, the bit-string that encodes nodes (joints) is used to virtually mask “invalid” edges (connections) for the purpose of fitness evaluation. However, the true value of those edges is retained during crossover and mutation so that there is no internal bias toward solutions with fewer edges.

3.2.2 Fitness criteria

The goal of our recognition algorithm is to find the mechanism topology graph that optimizes a series of objectives, which can be broadly categorized as relating to image consistency and mechanical feasibility. More specifically, we seek to maximize the image-based strength of joints and connections used in a solution, to minimize the strength of unused image information, and to maximize the likelihood that the mechanical behavior of the optimal solution is meaningful.

1. The *joint likelihood* (f_1) is the average detection strength

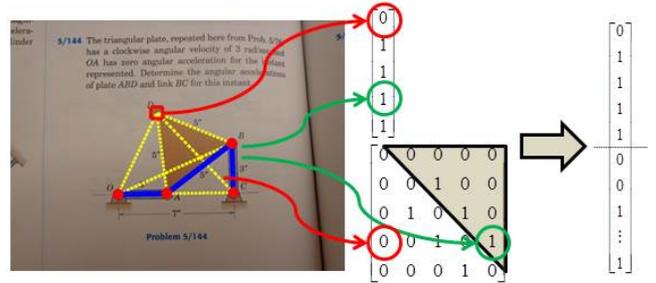


Figure 3: Abstracting the mechanism into a graph, then mapping the graph to a bit-string, where the first part of the chromosome identifies which joints exist in a solution, and the second part dictates the presence of connections between those joints.

of joints present in a given hypothesis (H), or

$$f_1(H) = \frac{\sum_{i=1}^N S_{j,i}}{\sum_{i=1}^N h_i} \quad (4)$$

where $S_{j,i}$ is the strength of the i th joint and h_i is the (Boolean) value of the i th gene.

2. Similarly, the *joint connection likelihood* (f_2) for a given hypothesis is computed as the average connection strength,

$$f_2(H) = \frac{\sum_{i=N+1}^L S_{c,i}}{\sum_{i=N+1}^L h_i} \quad (5)$$

where $S_{c,i}$ and L are the strength of the i th connection and the chromosome length, respectively.

3. *Residual image data* refers to the average strength of unused joints (f_3) and connections (f_4) in an individual solution. These two objectives are critical for enabling the detection of complex mechanisms; without them, a four-bar linkage (the simplest mechanism in our domain) will *always* be preferred unless additional nodes and edges improve the strength of f_1 or f_2 .
4. A series of binary *mechanical constraints* are evaluated to estimate the kinematic feasibility of solutions. The percentage of constraints that are satisfied make up the final objective (f_5); for a feasible mechanism,

this will have a value of 1. No constraint is more important than the others, and the feasibility objective contributes to nondominated sorting in the same way as the other fitness criteria. We chose this method (as opposed to approaches that always prefer feasible solutions [10, 4]) to allow infeasible, yet strong, solutions to persist because they are likely to be near the constraint boundaries.

While a full kinematic simulation would arguably provide the best insight regarding feasibility, it is simply too computationally expensive. Instead, we found through trial-and-error that simple heuristics yield a reasonable estimate of feasibility without the lag in computational speed. The current set of constraints used by our algorithm include: (i) the degrees of freedom³ (DOF) should be 1, (ii) each joint must have at least one connection, (iii) there must be at least four rigid bodies, including the frame, and (iv) all joints must be a minimum distance away from each other (*e.g.* 15-30 pixels).

3.2.3 Dealing with uncertainty

Uncertainty exists in the image consistency fitness criteria due to the nature of learned detector models and noise in the images. Comparing the distance to the SVM decision boundary for two joints may not always accurately reflect the true relative strength of those joints, especially when the relative distance is small. Similarly, minor illumination changes may induce small errors in the geodesic time calculations that dictate connection strength. These discrepancies can cause the true solution to be dominated by other solutions with similar, but unequal, strength. We implement two measures to combat this shortcoming. First, we add gaussian noise ($\mu = 0, \sigma = 0.02$) to the raw strength values on each generation. This increases the probability that two similar solutions remain on the same nondominated front, allowing “less fit” solutions to survive in the short term. Second, we do not require that the “best” solution be in the Pareto-optimal set; if the true solution exists in the population at the end of a run, there is a chance to recover it during post-processing.

3.2.4 Post-processing

The output of a typical multiobjective evolutionary algorithm is a set of Pareto-optimal solutions. In our application, however, only one solution⁴ is truly optimal. To reduce the final population in an effort to identify the true solution, we use the following procedure. First, all infeasible hypotheses are discarded. Next, duplicate solutions are identified via graph matching and subsequently removed. The remaining set is sorted based on a priority of objectives; currently, we prioritize connection strength because it seems to have greater discriminative power than joint detection.

4. EXPERIMENTS

4.1 MECH130 Dataset

³The *degrees of freedom* is the number of independent variables required to fully characterize the configuration of a mechanism.

⁴Or a very small set of solutions (see section 3.2.1)

Table 1: General NSGA-II parameters

Parameter	Symbol	Value
population size	μ	$200N^\dagger$
number of offspring	λ	μ
maximum number of generations	n	20
crossover method	–	uniform
crossover probability	p_c	0.9
mutation method	–	uniform
mutation probability	p_m	0.1
tournament size	k	0.05μ

[†] N refers to the number of detected joints

A custom dataset was created comprising 130 images from 5 textbooks [15, 22, 23, 24, 26] (see Figure 6 for examples). Each image shows a 1-DOF, closed, planar mechanism containing only revolute joints. No assumptions were made regarding position, scale, or orientation of the mechanism, although it is assumed the entire mechanism is included in the image. Images were not processed (*e.g.* cropping, filtering), so they may contain noise, illumination changes, and extraneous information such as text, annotations, pencil markings, or partial components from other mechanisms. The dataset is made publicly available as supplementary material to this paper, including hand-labeled ground truth information for each mechanism (joint locations and pairwise connections).

4.2 Methods

Two experimental studies were conducted on sample images from the MECH130 dataset to investigate the effectiveness of our approach. The first study aims to isolate the performance of the evolutionary algorithm from the vision aspects of the problem by creating artificial data for the image consistency fitness criteria based on generalized detection characteristics. The second study focuses on a practical implementation of the full system using a linear SVM over HOG features to locate joints and normalized geodesic time to compute likely connections, as described in section 3.1. Both experiments were performed using the same general settings listed in Table 1, and only differ in the ways outlined in the following paragraphs. The implementation has been developed in MATLAB [21], and all experiments were performed on an Intel(R) dual-core 2.50GHz CPU with 3GB RAM.

4.2.1 Parameterized Detection

Four images containing nonisomorphic mechanisms were selected for this study. All combinations of parameters values (Figure 2) were used to generate artificial detection strength data. For every 7-tuple, 5 random sets of strength values were generated, and 2 independent runs were executed on each set; this amounts to 116,640 total runs. Success rate is measured based on whether or not the true solution is found in the pruned and sorted population at the end of a run.

4.2.2 Vision-based Detection

A set of 15 images (3 per textbook) was hand-selected for testing. Considerable effort was made to include examples with varying rigid body abstraction, shape, and configuration as well as differing levels of extraneous noise. The remaining 115 images were used to train a linear SVM using

Table 2: Parameterization of detection performance

Parameter	Symbol	Values
1	Pr	{0.5, 0.7, 0.9}
2	$\Delta\mu_j$	{-0.4, -0.2, 0, 0.2}
3	$\sigma_{tp,j}$	{0.1, 0.2, 0.3}
4	$\sigma_{fp,j}$	{0.1, 0.2, 0.3}
5	$\Delta\mu_c$	{-0.2, 0, 0.2}
6	$\sigma_{tp,c}$	{0.1, 0.2, 0.3}
7	$\sigma_{fp,c}$	{0.1, 0.2, 0.3}

HOG features. Joint detection strength and rigid body likelihood were calculated according to the techniques in section 3.1. For each test image, 100 independent runs were performed using NSGA-II, and the results were pruned and sorted according to the procedure in section 3.2.4. See Figure 6 for an overview of this experiment.

4.3 Results and Discussion

4.3.1 Parameterized Detection

Figure 4 illustrates algorithm performance for the parameterized detection scheme described in previous sections. The goal of this study was to characterize the accuracy of our method in correctly identifying the underlying mechanical structure in an image for a generalized joint and rigid body detector. As expected, the success rate varies significantly depending on the parameters. In general, good performance is achieved when precision is high, the false positive joints are weaker than true positives, and the standard deviation of peak values remains low. It is clear from the figure that small changes in the parameters related to joint connection strength have the greatest impact on success rate. While the results may not look promising in this case, post-processing of our object detectors reveal that we are operating in the redder regions of the image. Figure 5 shows the normalized sensitivity of success rate to each of the 7 parameters and confirms that joint connection performance is critical to accuracy. Also, the mean peak of the false positives relative to true positives is important for both types of detection. The important outcome of this study is the ability to predict the performance of a detection scheme when you only know the general quality of the detector as defined by the parameterization.

4.3.2 Vision-based detection

An overview of this experiment is provided in Figure 6, and relevant performance measures are summarized in Table 3. The algorithm was able to find the correct solution at least once in 13/15 images. For those 13 solved images, the correct solution was found over 90% of the time. But perhaps the more important performance measure is *where* the solution was found. The top- N accuracy refers to the percentage of runs in which the true solution was at least in the top N solutions.

4.3.3 Limitations

Despite the promising results presented in this paper, our method has some limitations. For example, the algorithm operates under the assumption that all true joints are at least weakly detected; in the presence of a false negative, the method presented in this paper will never find the correct solution. Correcting this deficiency would require either

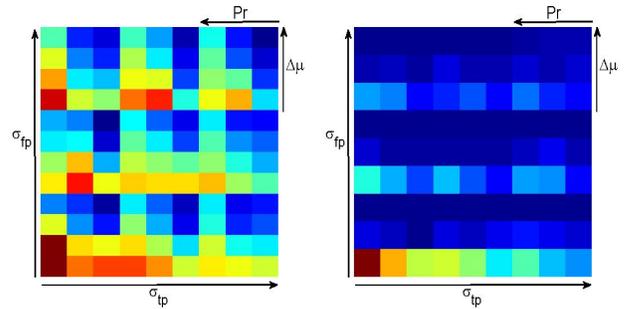


Figure 4: Success rate for parameterized tests on (a) joint detection and (b) joint connection likelihood. Each colored pixel represents the average success rate over all runs (1080 runs/pixel) for the subset of parameter vectors defined by the 4-tuple $\langle Pr, \Delta\mu, \sigma_{tp}, \sigma_{fp} \rangle$, normalized by the maximum accuracy. Blue values are low and red values are high.

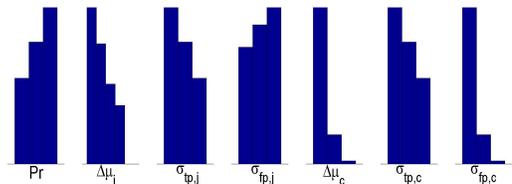


Figure 5: Normalized sensitivity of each parameter, as reflected by the average success rate for all corresponding runs.

a different representation of the hypothesis space or relaxation of joint detection thresholds. Another limitation is the narrow domain employed here; to fully assess how this algorithm generalizes, the scope of mechanisms must be increased.

5. CONCLUSIONS AND FUTURE WORK

Our approach leverages the well-understood computer vision methods for object recognition with the optimizing power of evolutionary computation. The efficacy of this method has been demonstrated on a series of textbook images and shown to solve more than 85% of problems, with the true solution being found in the top 5 optimal solutions 88% of the time. Another contribution of this work is the creation of a custom dataset comprising textbook images of planar mechanisms, which may serve as a benchmark for future approaches to solving this problem.

The algorithm has room for improvement in terms of accuracy, speed, and scalability. In the future, it is critical to investigate how well this method generalizes to more complex mechanisms (*e.g.* various joint types, spatial kinematics) and more difficult images (*e.g.* less abstract graphics, natural images of real systems); such robustness and scalability would make this a very practical computational tool for engineering design and analysis. For now, we have demonstrated a successful framework for mechanism recognition that can be easily adapted to new object detection schemes and optimization objectives.

Table 3: Vision-based experimental results

Performance Measure	Result
success rate, total	79.5%
success rate, solved images	91.8%
top-1 accuracy	50.9%
top-3 accuracy	79.2%
top-5 accuracy	88.0%
average time	4.11 ± 3.98 sec
average time per joint	0.57 ± 0.24 sec
average time per connection	0.219 ± 0.013 sec

6. ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant Nos. CMMI-0846730, CMMI-1031703, DUE-1043241, and GRFP-0946825. The authors kindly thank Aniruddha Basak for useful discussions on evolutionary algorithms.

7. REFERENCES

- [1] Adams. *version 2013.2*. MSC Software Corporation, Newport Beach, California, 2010.
- [2] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In *Computer Vision–ECCV 2006*, pages 404–417. Springer, 2006.
- [3] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. Brief: Binary robust independent elementary features. In *Computer Vision–ECCV 2010*, pages 778–792. Springer, 2010.
- [4] C. A. Coello Coello. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. *Computer methods in applied mechanics and engineering*, 191(11):1245–1287, 2002.
- [5] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(5):603–619, 2002.
- [6] D. Crandall, P. Felzenszwalb, and D. Huttenlocher. Spatial priors for part-based recognition using statistical models. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 10–17. IEEE, 2005.
- [7] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005.
- [8] R. Davis. Magic paper: sketch-understanding research. *Computer*, 40(9):34–41, 2007.
- [9] J. G. De Jalon and E. Bayo. *Kinematic and dynamic simulation of multibody systems*. Springer-Verlag New York, USA, 1994.
- [10] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *Evolutionary Computation, IEEE Transactions on*, 6(2):182–197, 2002.
- [11] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1627–1645, 2010.
- [12] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *International Journal of Computer Vision*, 61(1):55–79, 2005.
- [13] L. Fu and L. B. Kara. Recognizing network-like hand-drawn sketches: a convolutional neural network approach. ASME, 2009.
- [14] L. Fu and L. B. Kara. From engineering diagrams to engineering models: Visual recognition and applications. *Computer-Aided Design*, 43(3):278–292, 2011.
- [15] J. Ginsberg. *Engineering Dynamics*. Cambridge University Press, New York, New York, 2008.
- [16] M. Hegarty. Mental animation: inferring motion from static displays of mechanical systems. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 18(5):1084, 1992.
- [17] M. Hegarty. Mechanical reasoning by mental simulation. *Trends in cognitive sciences*, 8(6):280–285, 2004.
- [18] M. Hegarty and V. K. Sims. Individual differences in mental animation during mechanical reasoning. *Memory & Cognition*, 22(4):411–430, 1994.
- [19] H. Lipson. Evolutionary synthesis of kinematic mechanisms. *AI EDAM (Artificial Intelligence for Engineering Design, Analysis and Manufacturing)*, 22(3):195, 2008.
- [20] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.
- [21] MATLAB. *version 8.1.0.604 (R2013a)*. The MathWorks Inc., Natick, Massachusetts, 2013.
- [22] D. J. McGill and W. K. King. *Engineering Mechanics: An Introduction to Dynamics*. Tichenor Publishing, Bloomington, Indiana, 2003.
- [23] J. L. Meriam and L. G. Kraige. *Engineering Mechanics, Volume 2*. John Wiley and Sons, Inc., Singapore, 1993.
- [24] J. L. Meriam and L. G. Kraige. *Engineering Mechanics: Dynamics*. John Wiley and Sons, Inc., Hoboken, New Jersey, 2007.
- [25] P. Soille. Generalized geodesy via geodesic time. *Pattern Recognition Letters*, 15(12):1235–1240, 1994.
- [26] E. Soylemez. *Mechanisms*. Middle East Technical University, Ankara, Turkey, 1993.
- [27] N. Srinivas and K. Deb. Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary computation*, 2(3):221–248, 1994.
- [28] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–511. IEEE, 2001.
- [29] E. Zitzler and L. Thiele. Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach. *Evolutionary Computation, IEEE Transactions on*, 3(4):257–271, 1999.
- [30] M. Zitzler and Thiele. Spea2: Improving the strength pareto evolutionary algorithm, 2001.

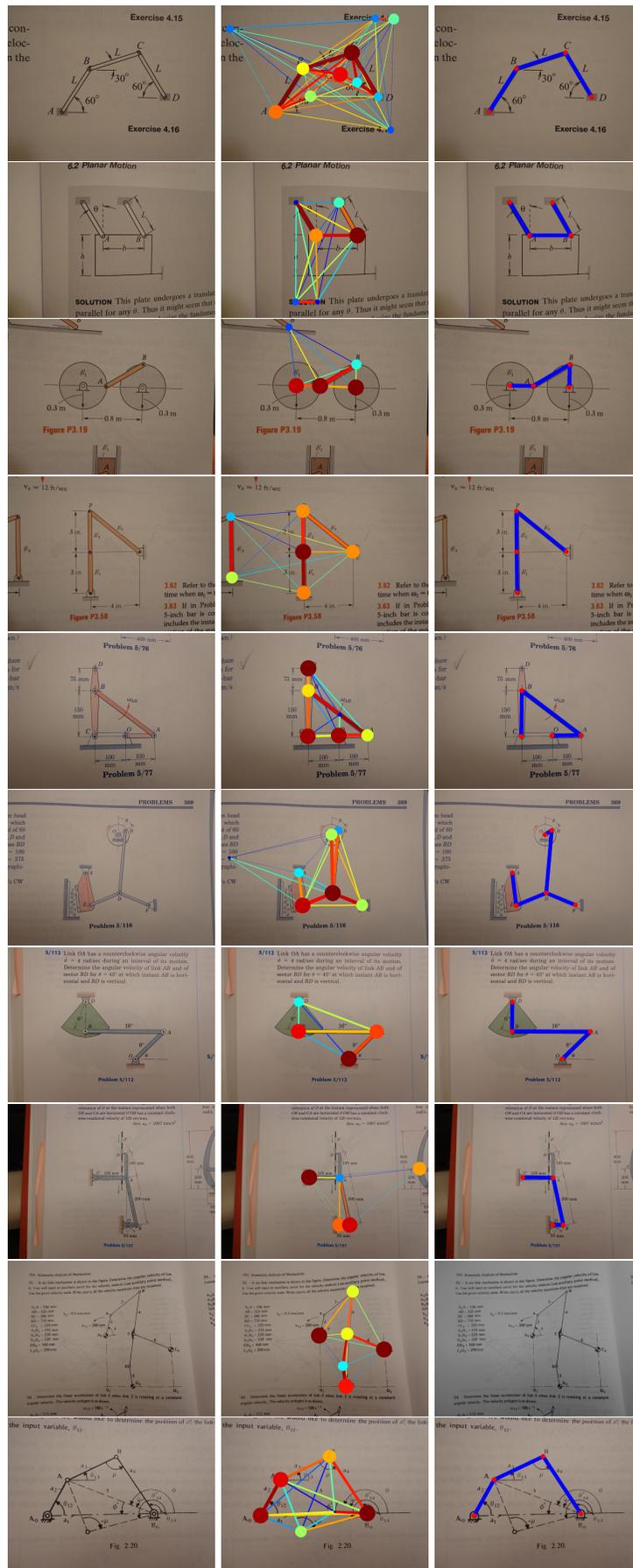


Figure 6: Example test images and results from MECH130 dataset. (a) Raw images. (b) Strength of joint detections and pairwise connections; higher values indicated by darker (red) color, thicker lines, and larger markers. (c) Graph representation of the ground truth (if it was ever found)