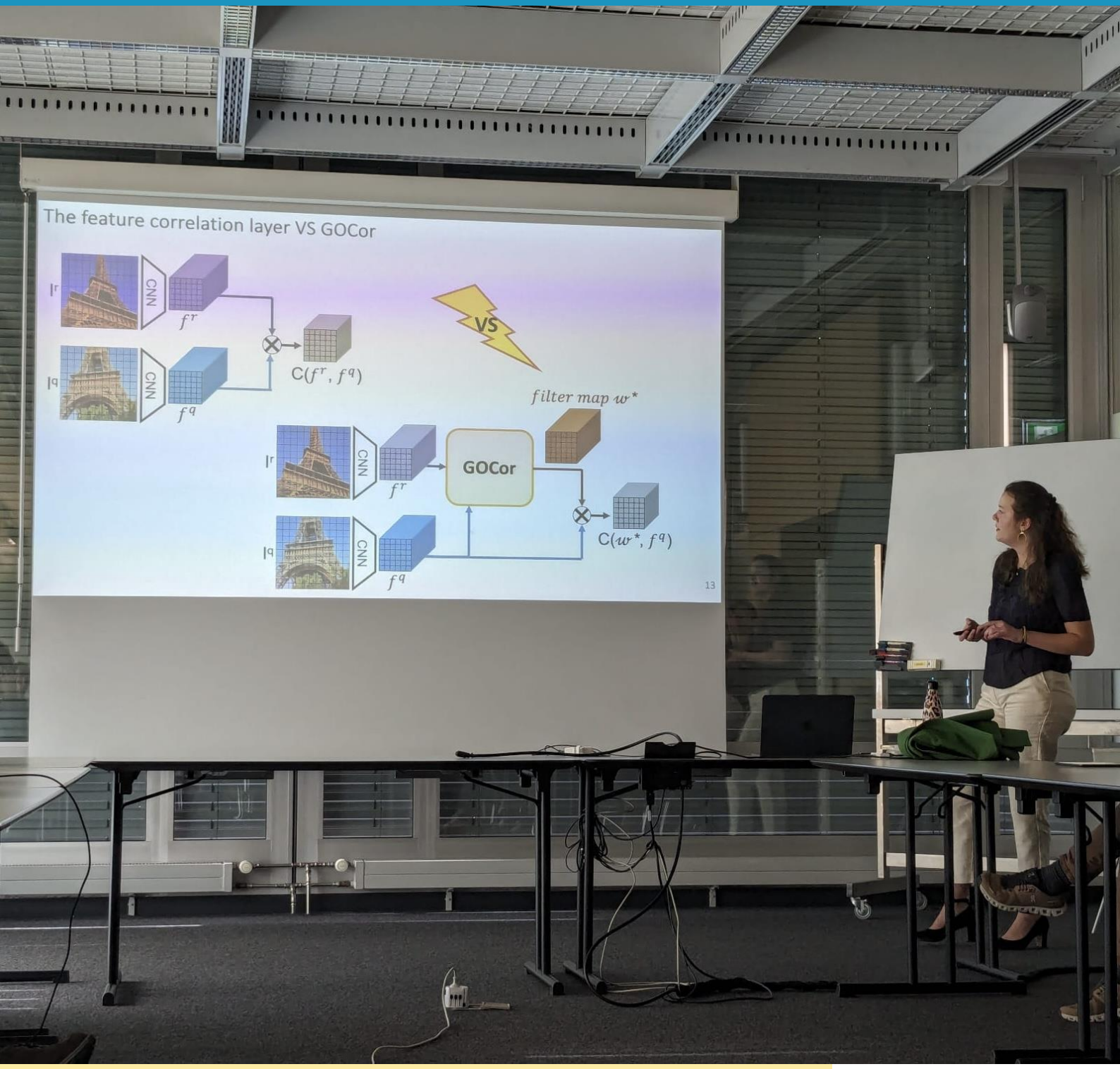
May 2024

Computer Vision News & Medical Imaging News

The Magazine of the Algorithm Community



A publication by RSIP VISION

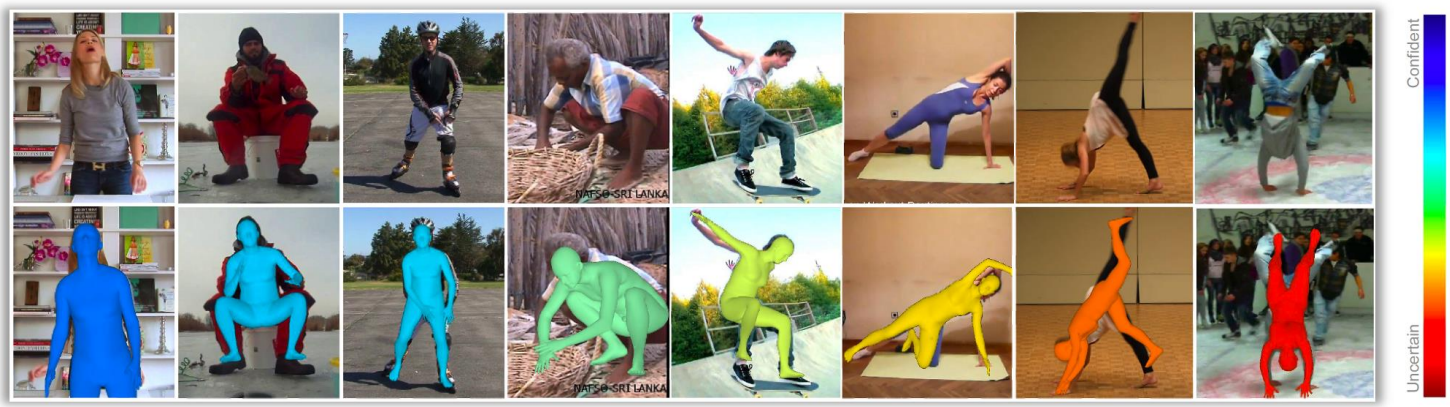
POCO: 3D Pose and Shape Estimation using Confidence

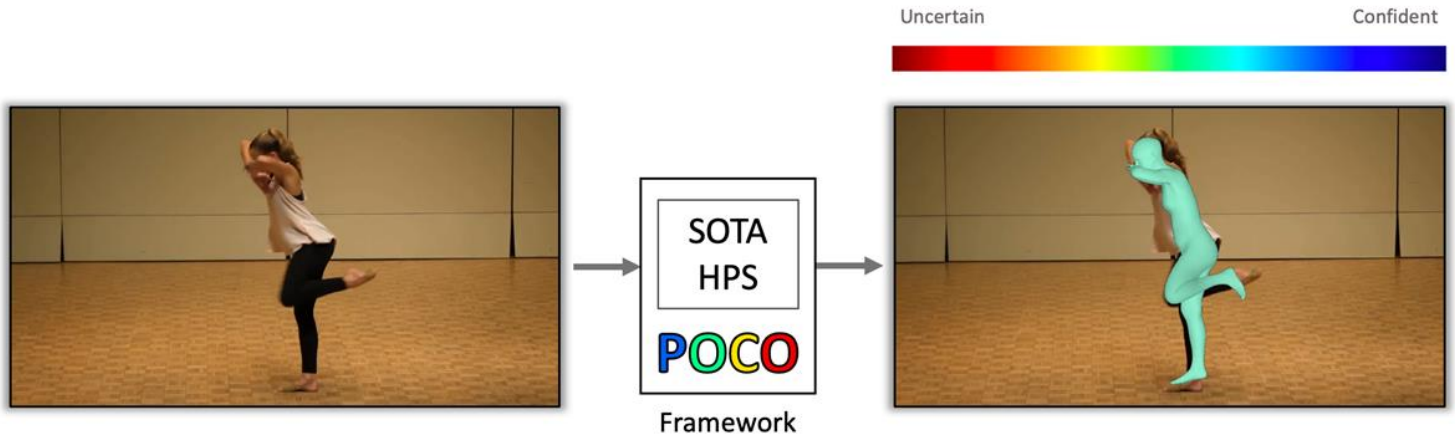
Sai Kumar Dwivedi (left) is a PhD student at the Max Planck Institute for Intelligent Systems MPI-IS, supervised by MPI-IS Director Michael J. Black and Dimitrios Tzionas (right), an assistant professor at the University of Amsterdam and formerly of MPI-IS, with which he continues to collaborate.

Sai and Dimitrios speak to us about their novel framework, POCO, which extends human pose and shape regressors to output both 3D bodies and uncertainty together.

Conventionally, human pose and shape regressors have focused solely on predicting a body's pose and shape without considering the confidence level associated with the output.

However, for downstream applications, **understanding the uncertainty of these predictions is crucial to ascertaining their reliability and usefulness**.



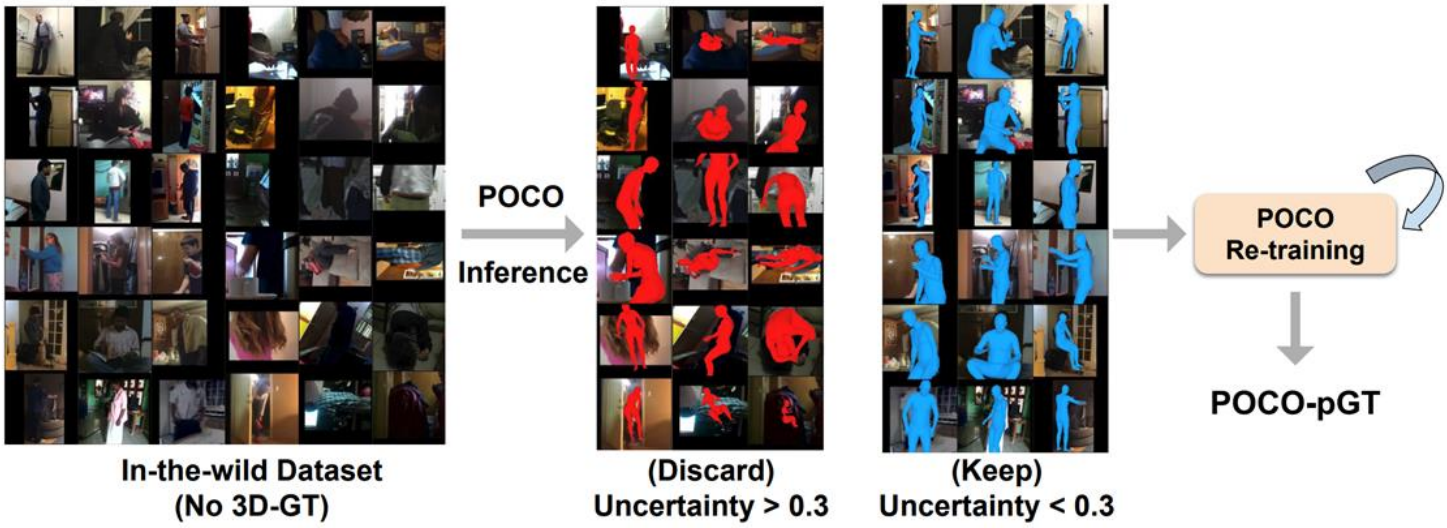


Previously, works on this topic commonly modeled uncertainty by treating the predicted poses and shapes as probability distributions, **calculating confidence by sampling multiple instances of the predicted distributions, and then analyzing the variability or deviation among these samples**. *"However, this approach has a problem,"* Sai tells us. *"First, it's very slow because you have to do multiple samples to get the uncertainty estimate. Second, there's a trade-off between speed and accuracy."*

POCO can convert any state-of-the-art human pose and shape regressor into an approach that estimates the uncertainty quickly and accurately in a single forward pass. *"The important thing is it doesn't hurt the accuracy of the pose estimation,"* Dimitrios confirms. *"POCO gives you an extra modality as output, which is the confidence of the estimation, but the accuracy of the pose estimation itself is not compromised."*

In fact, in a recent LinkedIn post, **Michael Black** revealed that POCO improves the method's accuracy. *"An interesting byproduct is that we find that **training methods to estimate their certainty actually makes them more accurate** – the effect is small but consistent across all regressors tested."* He added: *"There is only upside to using POCO."*

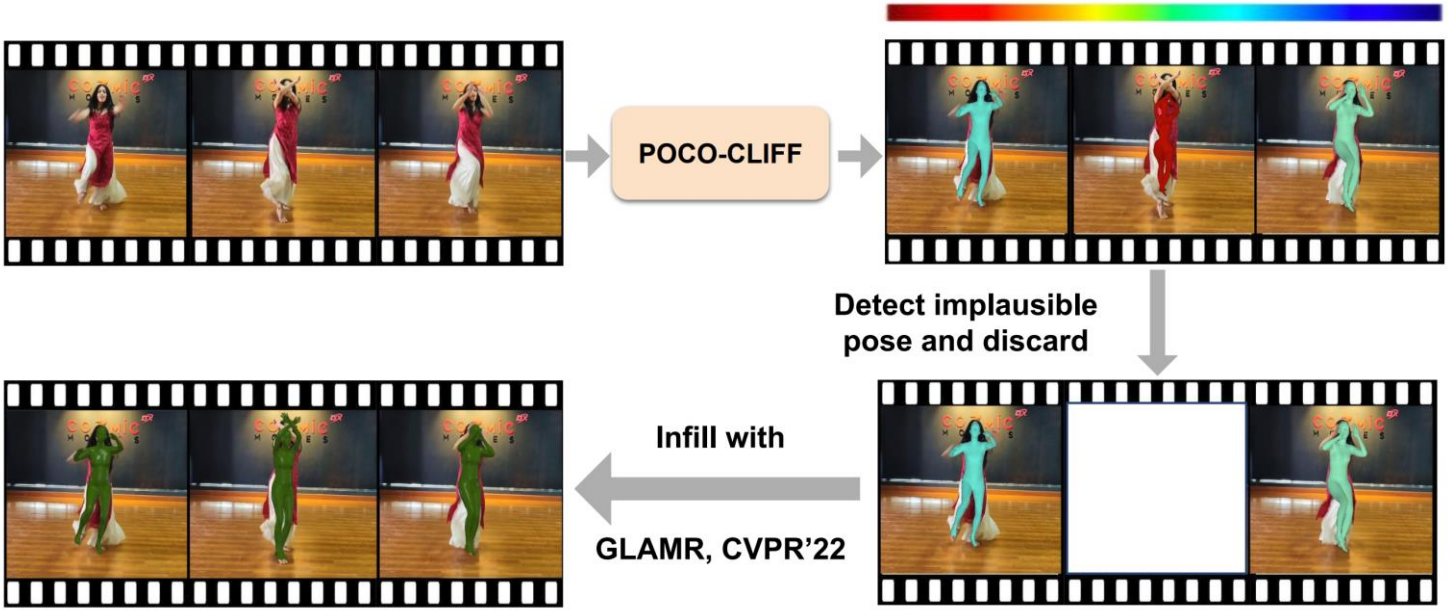
Nevertheless, the journey toward integrating confidence with pose estimation has not been without its challenges. Supervising uncertainty where there is no ground truth is very complex, and convincing people that it is possible to use it without affecting performance can be difficult. *"The computer vision community is laser-focused on benchmarks,"* Dimitrios points out. *"We want accuracy to be better, but sometimes we forget about the useful context around it. We want our methods to be very accurate, but **we also need to know when to trust them!**"*



Versatility is at the heart of POCO. "*Our framework is agnostic to any other method that estimates human pose,*" Sai explains. "*For modeling uncertainty, we use conditional normalizing flow, a technique that takes a simpler distribution and converts it into a complex distribution.*"

POCO's stellar team credits its success to a collaborative ethos, strategic partnerships fostered through initiatives like the **European Laboratory for Learning and Intelligent Systems (ELLIS)**, and the support of organizations like the **German Federal Ministry of Education and Research (BMBF)**.

Along with forging strong ties with several ELLIS units around Europe, the team continues collaborating with INRIA Research Director **Cordelia Schmid**, a longtime MPI-IS friend. "*Doing good research takes a village,*" Dimitrios attests. "*It's very hard for isolated individuals to*

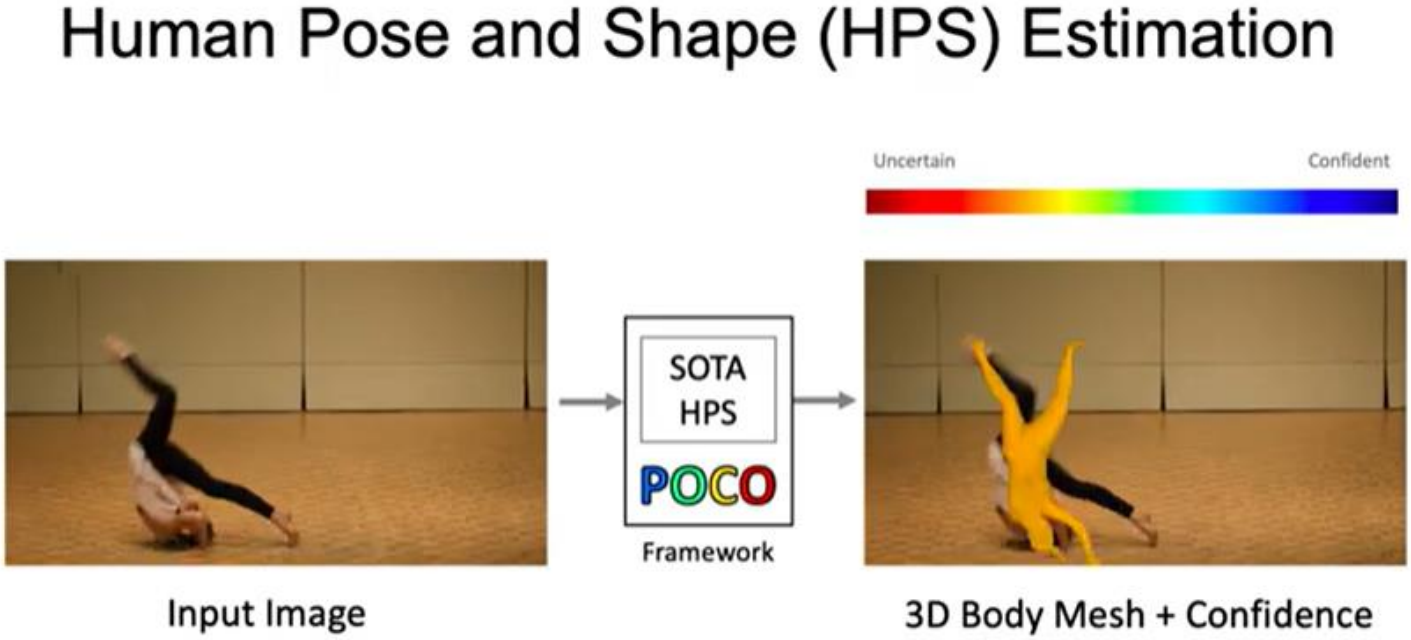


up with novel ideas and a useful research outcome."

There are several possibilities for future work exploring shape and motion uncertainty in greater depth. POCO estimates the pose of a single body performing some motions, but **the more ill-posed the task, the more useful the confidence estimation**. When the body starts moving and interacting with other objects and people and gets occluded, this extra uncertainty signal will really shine.

Sai and Dimitrios are adamant that **POCO is for everyone**. It comes at no cost, is easy to train, and generalizes to different architectures. They hope estimating confidence will become standard practice in computer vision, enhancing how we perceive and trust machine learning algorithms. "*Critical applications, such as medical tasks, use uncertainty a lot, but I don't see why computer vision can't use it as an additional modality,*" Sai states. "*People model uncertainty in various ways. When talking about classification, they use logits, but that's not accurate. I don't see why computer vision couldn't have this as a standard way of representing any output.*"

POCO has the potential to be part of **a new era of more reliable and trustworthy AI systems**. "*An experienced surgeon is trusted way more than a beginner due to the confidence in their actions – we've tried to instill this in our method,*" Dimitrios adds. "*Our code is online, so we'll be the happiest people on the planet if people integrate POCO into their work!*"



Vision-aided Screw Theory-based Inverse Kinematics Control of a Robot Arm Using Robot Operating System (ROS2) - Part 3

Do you remember awesome Madi Babaiasl, who two years ago promptly taught me about redundancy in Robotics? A really fascinating moment!

We asked her to prepare a full lesson for our readers. What she did with three of her students (they are all PhD students and one of them is a soon-to-be professor) is a great "educational" lesson – so complete that we had to divide it in 3 parts. Parts 1 and 2 were published in the March and April issues of Computer Vision News.

Here is part 3 of the fist lesson.

by Madi Babaiasl, Bryan MacGavin, Daniel Montes Tolon, Namrata Roy

1 Introduction

Up to this point, you should have acquired an understanding of setting up your hardware and software environment and delved into the theoretical aspects of screw theory-based numerical inverse kinematics using the Newton-Raphson iterative method. With the groundwork laid, we are now ready to move forward into the practical implementation phase, where you will begin to apply the vision-aided numerical inverse kinematics control to the robot arm. By the end this lesson, you will be able to implement the numerical inverse kinematics of your robot arm in Python, get feedback

from the camera to find the location of the clusters (objects) and then feed the 3D position of the objects to the inverse kinematics code to be used as the desired position. You will then be able to use ROS to command the robot joints to move the end-effector to this desired position. Feel free to experiment and design your own project.

2 Vision-Aided Numerical Inverse Kinematics Control of the Robot Arm

The objective of this part is to merge the numerical inverse kinematics from the previous part with the robot's perception package to create a vision-aided inverse kinematics mission planner. The depth camera will first capture the AprilTag attached to the robot's arm to help ROS figure out the homogeneous transformation of the robot's base frame relative to the camera and vice versa. This will help convert the camera's depth readings of scene objects to homogeneous transformations with respect to the robot's base. Later, the inverse kinematics module will map these transformations to joint angle set-points to make the robot catch and release these objects. For a short introduction to image processing basics, you can refer to this lesson.

The camera that we used is the Intel D415 RealSense Depth Camera, which features 1920×1080 resolution at 30fps with a 65o × 40o field of view and an ideal range of 0.5m to 3m. Using the stereo vision technology and the infrared sensor will help us to estimate the 3D structure of the scene.

2.1 Physical Setup and Running the Perception Pipeline

First, you need to make the physical setup ready. To begin, construct your stand and fasten the RealSense camera onto the 1/4-inch screw located at the top. Subsequently, position the stand in your designated workspace and manipulate the goose-neck or ball/socket joint to orient the camera towards your tabletop. Then, arrange your target objects and the robot in a way that the objects and the AprilTag are clearly visible in the camera's view. Fig.1 shows a sample setup (for your reference).

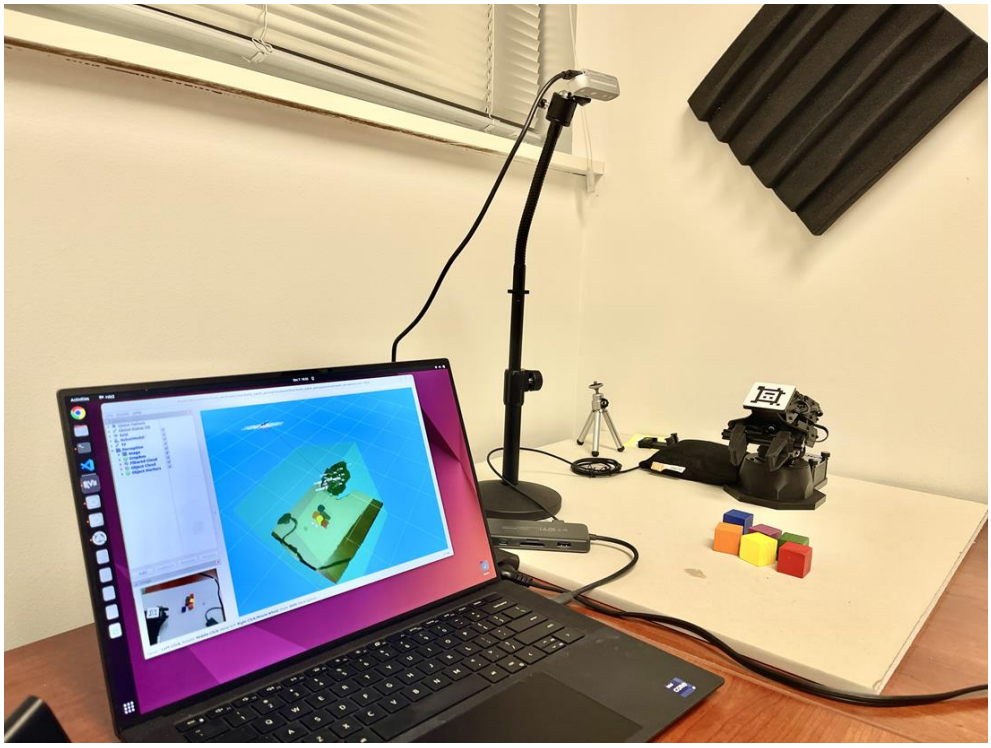


Fig. 1: Experimental Setup. This setup includes a computer running ROS2 with RViz on Ubuntu 22.04, the camera on stand, the robot arm, and the objects.

There is nothing special about this setup. Just set up the camera in a way that the AprilTag and the objects are clearly visible. Also, choose objects that can fit into the robot's gripper and avoid reflective ones, as they may disrupt the depth camera's ability to detect them using infrared light. Also, the scene cannot have direct sunlight present for the same reason.

For the robot to detect the position of each object and pick them, first, it is necessary for us to know the position of the camera relative to the arm. We can do this by manually measuring the offset between the camera's color optical frame and the robot's 'base_link' . However, using this method is extremely time-consuming and prone to errors. Instead, we utilize the apriltag_ros ROS2 package to determine the transformation of the AprilTag visual fiducial marker on the arm's end-effector relative to the camera's color optical frame. Afterward, the transformation from the camera's color optical frame to the arm's 'base_link' frame is computed and then published as a static transform.

To run the perception pipeline, run the following launch command in a terminal (note that you should first install the ROS2, and Python-ROS API from previous parts):

```
ros2 launch interbotix_xsarm_perception xsarm_perception.launch.py robot_model:=
    px100 use_armtag_tuner_gui:=true use_pointcloud_tuner_gui:=true
```

Here, you can play around with the GUI a bit to get familiar with the perception pipeline (we will code everything and will not use the GUI, but it is good to get familiar with it). The first command is *use_armtag_tuner_gui:=true*, which will open up a Graphical User Interface (GUI) that will allow you to figure out where the position of the arm is with respect to the camera. In the Armtag tuner GUI, you can see a field named *'num samples'*. This can vary from 1 to 10, with 10 indicating the highest level of accuracy. When you press the *'snap pose'* button, the system will capture the specified quantity of images as defined in the *'num samples'* setting. The AprilTag algorithm will then run on each of the images, probably producing slightly different poses for the AprilTag's location relative to the camera. Finally, the program behind the GUI will average all those poses to hopefully obtain the most accurate position. As you will see in the GUI message the snapped pose represents the transform from the *'camera_optical_frame'* frame to the *'px_100/base_link'* frame. Try to verify this by manually measuring the portion of the camera w.r.t the base.

At this point, you should see a pointcloud version of your tabletop with the objects on it. The second GUI applies some filters on the image in a way that all the objects are clear in our image. This GUI obtains the raw point cloud from our depth camera and applies several filters to it in a manner that makes the objects (clusters) visible. This GUI contains several sections for filtering the point cloud. The description for each of these sections is provided in the GUI windows, and you can also see this guide for how to go about doing this. In general, this GUI will employ a mathematical model to identify the plane on which the objects are positioned. Then, it applies a Radius Outlier Removal filter to omit 'noisy' points in the point cloud. To understand how to create a perception pipeline, you can follow the instructions in this link. Now, let's go ahead and implement our own code for cluster detection using the camera and picking and placing by solving the numerical inverse kinematics of the arm at the desired poses detected by the camera.

2.2 Python Implementation of Numerical Inverse Kinematics of the Robot Arm

To implement vision-aided inverse kinematics, we need to first implement the numerical inverse kinematics of the robot arm in ROS2. For implementation of this part in ROS2, you should start with updating the configuration yaml files to set the robot motion mode to 'position.' Following that, you should create the helper functions that will do all the calculations and algorithm implementations for you. Finally, you will create your custom APIs.

2.2.1 Setup and Helper Functions

First off, change the yaml file content to 'position' mode.

After that, create a package, and then create a new script called px100_IK.py (or any appropriate name you choose). If you do not know how to create a package in ROS2, follow the instructions in this link. In this script, develop a main function and a special class for inverse kinematics. We called this class as ourAPI. The code below gives you a template to start this:

```
from math import atan2, sqrt, pi, acos, sin, cos, asin
from scipy.linalg import logm, expm
import numpy as np

class ourAPI:
    def __init__(self):
        # Robot parameters
        self.L1 = 0.08945
        self.L2 = 0.1
        self.Lm = 0.035
        self.L3 = 0.1
        self.L4 = 0.08605
        self.S = np.array([[0, 0, 1, 0, 0, 0],
                           [0, 1, 0, -0.08945, 0, 0],
                           [0, 1, 0, -0.18945, 0, 0.035],
                           [0, 1, 0, -0.18945, 0, 0.135]]) # Screw axes
        self.M = np.array([[1, 0, 0, 0.22105],
                           [0, 1, 0, 0],
                           [0, 0, 1, 0.18945],
                           [0, 0, 0, 1]]) # End-effector M matrix
```

You will need to start writing the class helper functions to implement the numerical inverse kinematics and adding them to the class

methods one by one. As we saw, The numerical method relies on the forward kinematics (homogeneous transformation of the end-effector with respect to the base frame Tsb). This requires screw axes assignments from previous parts and computing the exponential form of the transformation matrix that we also had in previous parts of this lesson. Therefore you need to create the function that will convert a combination of a screw axis and joint angle to a homogeneous transformation matrix (exponentiation), as follows:

```
def screw_axis_to_transformation_matrix(self, screw_axis, angle):
    """
    Convert a screw axis and angle to a homogeneous transformation matrix.

    Parameters:
    - screw_axis: A 6D screw axis [Sw, Sv], where Sw is the rotational component
                  and Sv is the translational component.
    - angle: The angle of rotation in radians.

    Returns:
    - transformation_matrix: The 4x4 homogeneous transformation matrix
                             corresponding to the input screw axis and angle.
    """
    assert len(screw_axis) == 6, "Input screw axis must have six components"

    # Extract rotational and translational components from the screw axis
    Sw = screw_axis[:3]
    Sv = screw_axis[3:]

    # Matrix form of the screw axis
    screw_matrix = np.zeros((4, 4))
    screw_matrix[:3, :3] = np.array([[0, -w[2], w[1]],
                                     [w[2], 0, -w[0]],
                                     [-w[1], w[0], 0]])
    screw_matrix[:3, 3] = Sv

    # Exponential map to get the transformation matrix
    exponential_map = expm(angle * screw_matrix)

    return exponential_map
```

In the numerical method, you need to get the twist vector from the matrix form of the twist vector. The following helper function does this job for you:

Computer Vision News is very grateful to Madi and her team for this awesome lesson in robotics!

```
def twist_vector_from_twist_matrix(self, twist_matrix):
    """
    Compute the original 6D twist vector from a 4x4 twist matrix.

    Parameters:
    - twist_matrix: A 4x4 matrix representing the matrix form of the twist

    Returns:
    - twist_vector: The 6D twist vector [w, v] corresponding to the input
                    twist matrix.
    """
    assert twist_matrix.shape == (4, 4), "Input matrix must be 4x4"

    w = np.array([skew_symmetric_matrix[2, 1], skew_symmetric_matrix[0, 2],
    skew_symmetric_matrix[1, 0]])
    v = skew_symmetric_matrix[:3, 3]

    return np.concatenate((w, v))
```

In the numerical algorithm, you need to compute the body Jacobian of the robot iteratively. Let's create a Jacobian function that takes in the arm angles and gives out the body Jacobian.

```
def body_jacobian(self, angles):
    # Calculate the body jacobian
    J = np.array([[-sin(angles[1]+angles[2]+angles[3]), 0.0, 0.0, 0.0],
                  [0.0, 1.0, 1.0, 1.0],
                  [cos(angles[1]+angles[2]+angles[3]), 0.0, 0.0, 0.0],
                  [0.0, self.L3*cos(angles[2])*sin(angles[2]+angles[3])+self.L4*
    cos(angles[2]+angles[3])*sin(angles[2]+angles[3]), self.L3*sin(angles[3])+self.L4
    *sin(angles[3])*cos(angles[3]), 0.0],
                  [self.Lm*cos(angles[1])+self.L2*sin(angles[1]), 0.0, 0.0, 0.0],
                  [0.0, -self.L3*cos(angles[2])*cos(angles[2]+angles[3])-self.L4*
    cos(angles[2]+angles[3])**2, -self.L3*cos(angles[3])-self.L4*cos(angles[3])**2, -
    self.L4]])
    return J
```

2.2.2 Algorithm implementation

The numerical algorithm explained in the previous part of this lesson can be implemented as follows:

```
def num_IK(self, Tsd, InitGuess):
    """
    Gives joint angles using numerical method.
    """
    for i in range(1000):
        # Calculate the end-effector transform (Tsb) evaluated at the InitGuess
    using the helper functions that you wrote at the beginning.
        Tsb = self.screw_axis_to_transformation_matrix(self.S[0,:], InitGuess[0]) @
    \
            self.screw_axis_to_transformation_matrix(self.S[1,:], InitGuess[1]) @ \
            self.screw_axis_to_transformation_matrix(self.S[2,:], InitGuess[2]) @ \
```

```
            self.screw_axis_to_transformation_matrix(self.S[3,:], InitGuess[3]) @
    self.M

        # Compute the body twist
        matrix_Vb = logm(np.linalg.inv(Tsb)@Tsd); Vb = self.
    twist_vector_from_skew_symmetric_matrix(skew_sym_Vb) # use the helper function at
     the beginning to extract the vector

        # Compute new angles
        NewGuess = InitGuess + np.linalg.pinv(self.body_jacobian(InitGuess))@Vb
        print(f"Iteration number: {i} \n")

        # Check if you're done and update initial guess
        if(np.linalg.norm(abs(NewGuess-InitGuess)) <= 0.001):
            return [NewGuess[0], NewGuess[1], NewGuess[2], NewGuess[3]]
        else:
            InitGuess = NewGuess
    print('Numerical solution failed!!')
```

By now, you have completed the numerical inverse kinematics implementation. In the next subsection, we will add vision feedback and implement vision-aided inverse kinematics. For now, save this Python file as it will be needed for the next part.

2.3 Vision-aided Inverse Kinematics Code Implementation

By now and for the previous part, your should have changed the yaml file content to 'position' mode, if you have not done so, change the robot settings by copying the following to the yaml file as follows:

```
groups:
 arm:
  operating_mode: position
  profile_type: time
  profile_velocity: 2500
  profile_acceleration: 300
  torque_enable: true

singles:
 gripper:
  operating_mode: PWM
  torque_enable: true
```

You also have the inverse kinematics script from previous part, which we will use for this part as well. Here, we will use the camera's feedback to detect objects and estimate those desired poses, and then use the inverse kinematics to move the robot to the position of these objects and pick and place them.

After we developed our low-level API's in previous part, now we need to write down a new script that contains our mission planner. Let's call this script px100_vision_IK (or any name that you like). We will import our custom-defined library in this file, as well as the vision module and some other basic libraries, by adding the following imports:

```
from interbotix_perception_modules.armtag import InterbotixArmTagInterface
from interbotix_perception_modules.pointcloud import InterbotixPointCloudInterface
from interbotix_xs_modules.xs_robot.arm import InterbotixManipulatorXS
from IK import ourAPI
import numpy as np
from math import atan2, sin, cos, pi
# You can use the time library if you ever need to make some delay. For example:
    time.sleep(3)
import time
```

We now need to define some constants required for the vision module operation as well as the standard homogeneous transformation for the basket (or any object based on your design) pose. Remember that the vision module needs to define a number of reference frames: the camera's reference frame (the vision module references the 3D coordinate data with respect to the camera's optical center), the arm tag reference frame (the frame of the AprilTag attached to the robot arm, which is captured by the camera in the very beginning to determine where the end-effector stands from the camera's optical center), and finally the robot's base frame (which is required to reference everything with respect to the robot's base frame instead of the camera's optical center). This can be done as follows:

```
# Start by defining some constants such as robot model, visual perception frames,
    basket transform, etc.
ROBOT_MODEL = 'px100'
ROBOT_NAME = ROBOT_MODEL
REF_FRAME = 'camera_color_optical_frame'
ARM_TAG_FRAME = f'{ROBOT_NAME}/ar_tag_link'
ARM_BASE_FRAME = f'{ROBOT_NAME}/base_link'
```

We now have all the global variables and dependencies that we need to design our mission planner; it is time to create our main function and start to create a robot object, a cloud interface object (for 3D point cloud and depth data generation), an arm tag interface object (for AprilTag identification), and an inverse kinematics object:

```
def main():
    # Initialize the arm module along with the point cloud, armtag modules and
    px100_IK_ex custom API
    bot = InterbotixManipulatorXS(
        robot_model=ROBOT_MODEL,
        robot_name=ROBOT_NAME,
        group_name='arm',
        gripper_name='gripper'
    )
    pcl = InterbotixPointCloudInterface(node_inf=bot.core)
    armtag = InterbotixArmTagInterface(
        ref_frame=REF_FRAME,
        arm_tag_frame=ARM_TAG_FRAME,
        arm_base_frame=ARM_BASE_FRAME,
        node_inf=bot.core
    )
    my_api = ourAPI()
```

In the beginning, we need to make sure the arm's gripper is in the release position and that we start from the sleep pose. Append the following lines to the main code:

```
    # set initial arm and gripper pose
    bot.arm.go_to_sleep_pose()
    bot.gripper.release()
```

The robot arm AprilTag should now be in the field of view of the camera, so now we attempt to solve the problem of finding the homogeneous transformation of the base with respect to the camera frame. The point cloud object would later use this information to reference the cubes to the robot base frame directly. Append the following lines to the main code:

```
    # get the ArmTag pose
    armtag.find_ref_to_arm_base_transform()
```

Now, it is time to get the homogeneous transformations of the cubes (clusters) relative to the robot base frame using the .get_cluster_positions method. The method will sort the cubes based on how far they are along the x-axis of the base frame. Append the following lines to the main code:

```
    # get the cluster positions
    # sort them from max to min 'x' position w.r.t. the ARM_BASE_FRAME
    success, clusters = pcl.get_cluster_positions(
        ref_frame=ARM_BASE_FRAME,
        sort_axis='x',
        reverse=True
    )
```

Create bounds for the RGB values of the clusters. The camera would surely pick unwanted objects such as wires, stains, etc., as clusters (you can verify this by running your code in debug mode and seeing the variables for clusters on the left-hand side where you will see that there are other clusters than the object that you intend to pick up). Therefore, we need to consider only some colors. In this example, we use a blue cube and, therefore, pick RGB thresholds for different shades of blue. You repeat this for other colors of clusters in your scene.

Note: The clusters have a color property; you can see those by putting breakpoints after the above code and looking at the cluster variables on the left-hand side to see their RGB values. Those can also give you an idea of the color space that the camera is seeing.

```
# Create a blue color bound
    # Define a range for blue color
    lower_blue = (0, 0, 15)
    upper_blue = (55, 75, 255)
```

Now, it is time to write down our mission planner. We need to loop over the clusters/cubes one at a time and do the following in order:

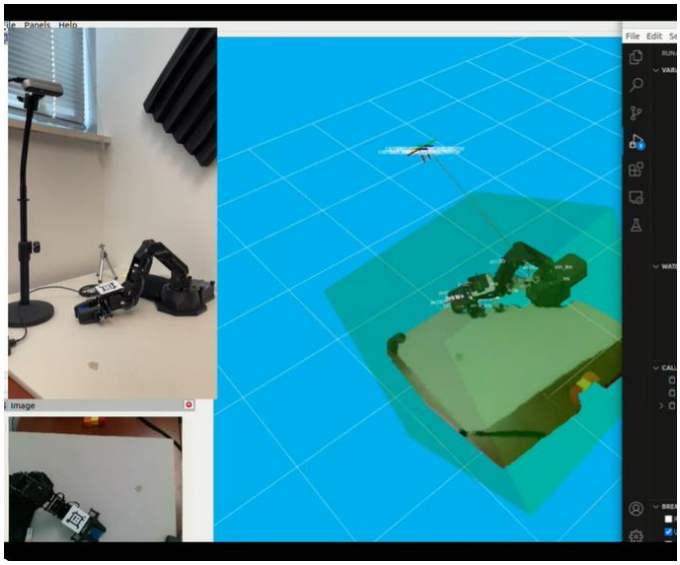
1. Determine the cube's (x,y) coordinate and add a slight offset to the z coordinate to avoid pushing the cube away while approaching.
2. Adjust the end-effector orientation so that its x-axis is perfectly aligned with the horizontal axis (in other words, the end-effector frame should point forward). The first joint (waist) should be rotated by the angle θ, where θ could be computed as follows: θ = atan2(x, y), where x, and y are the detected cluster position x and y coordinates. This will make sure that the end-effector is oriented towards the object. **NOTE: This is again our solution. Your experiment may need other adjustments.**
3. The end-effector will go down to hold the cube and then grasp it.
4. The end-effector will go back to the original slightly higher position to avoid hitting other cubes on its way to the basket.

```
    if success:
        bot.arm.go_to_home_pose()
        # pick up all the objects and drop them in a basket (note: you can design
    your own experiment)
        for cluster in clusters:
            if all(lower_blue[i] <= cluster['color'][i] <= upper_blue[i] for i in
    range(3)):
                # Get the first cube location
                x, y, z = cluster['position']; z = z + 0.05 # Fingers link offset (
    change this offset to match your experiment)
                print(x, y, z)

                # Go on top of the selected cube
                theta_base = atan2(y,x) # Waist angle offset
                new_x = x/cos(theta_base)-0.01 #adjust this also to your own
    experiment. This is just an example.
                # desired pose
                Td_grasp = np.array([[1,   0,   0,   new_x],
                        [0,   1,   0,   0],
                        [0,   0,   1,   z],
                        [0,   0,   0,   1]])

                joint_positions = my_api.num_IK(Td_grasp, np.array([0,0,0,0])) #
    Numerical inverse kinematics

                # Here, I rotated the waist by theta_base. This positioned my end-
    effector towards the cluster.
                bot.arm.set_joint_positions(np.append(theta_base,joint_positions
    [1:])) # Set position
    else:
        print('Could not get cluster positions.')
```

Finally, you could go to the sleep pose. Don't forget to add the main run line as follows:

```
    # Go to sleep
    bot.arm.go_to_sleep_pose()
    bot.shutdown()


if __name__ == '__main__':
    main()
```

If you have followed the instructions accurately, your implementation should resemble the one depicted in the video on the right. Keep in mind, this is just an example, and you're encouraged to tweak the code to suit the needs of your specific experiment.



2.4 Possible Challenges

Here are possible challenges that you may encounter while implementing this lesson:

- Object Detection Relies on Color: Changes in lighting can affect color perception, which may require adjusting the defined color ranges. You can run your code in the debug mode and put a breakpoint right after getting the clusters, and in the variables, see the clusters and their color property.
- Visibility of AprilTag and Object: Both must be clearly visible to the camera for accurate detection and positioning.
- Handling IK Failures: The script anticipates potential failures in the numerical solution of the IK, indicating either unreachable poses or failure in detecting clusters.

2.5 More on Image Processing: Deep Learning

So far, we have handled the Apriltag processing by using conventional approaches. These conventional approaches rely on some handcrafted features by experts. However, as will become apparent in your trials, handcrafted features often produce errors when the environment slightly changes.

Slight changes in light intensity might affect the color thresholding, leading to the identification of the wrong cubes. The camera will not be able to perfectly capture the clusters if you're sitting in a dark room. Deep neural networks can automatically learn hierarchical representations and features from raw data. This allows them to adapt to the inherent complexity and variability in images, potentially uncovering intricate patterns that may be challenging for manual feature engineering.

This is why modern literature relies heavily on deep learning techniques. The detection problem is basically treated as an optimization problem, where engineers feed huge amounts of

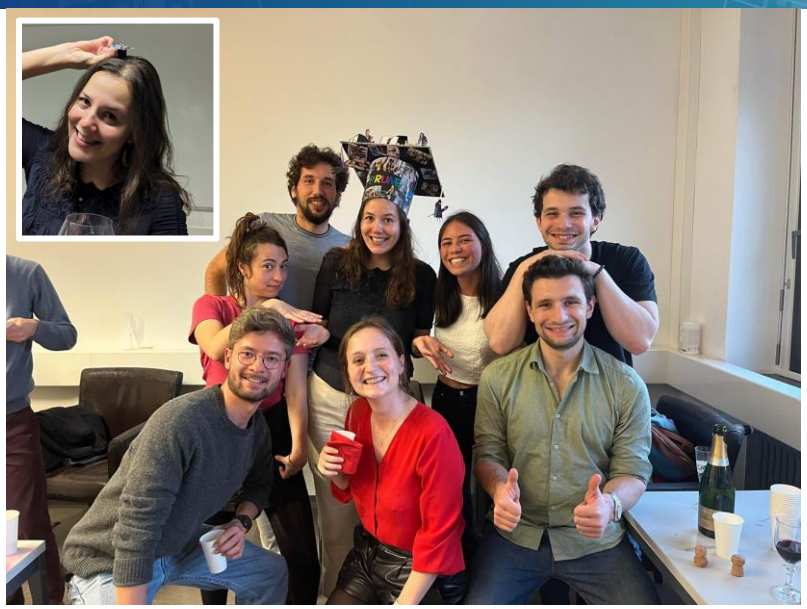
examples (called training datasets) to a neural network. The neural network uses these training examples to learn new features that are hard to handcraft. One good example is the YOLO (You only look once) pipeline, which can be trained to detect virtually any object, including cars, pedestrians, animals, and so on. In fact, YOLO itself has so many versions that each version surpasses its predecessor in accuracy and speed.

3 Summary

These lesson series presented a comprehensive guide on implementing vision-aided screw theory based inverse kinematics control for a robot arm using ROS2. Starting with an introduction to screw theory and its application in robotic inverse kinematics, these lessons detailed the setup and configuration process for the hardware and software including the robot arm, the vision kit, ROS2, RViz, and Python-ROS API. The core of the discussion revolved around the development and implementation of numerical inverse kinematics solutions. Using the Newton-Raphson iterative method, we described a systematic approach to solving the inverse kinematics problem, providing clear algorithms and code examples. Furthermore, the lessons delved into the integration of vision systems with ROS2 to enable object detection and manipulation tasks. By capturing the AprilTag attached to the robot's arm, the system calculated the transformation between the robot's base frame and the camera, facilitating the conversion of camera depth readings into homogeneous transformations.

This process allows for the precise manipulation of objects based on visual feedback. Several possible challenges, such as object detection reliance on color and visibility issues of AprilTag, are discussed, alongside solutions and best practices for troubleshooting and optimization. The lessons concluded by highlighting the limitations of traditional perception methods and the potential of deep learning techniques, specifically mentioning the YOLO pipeline for enhanced robotic vision.

References: see here



Prune Truong recently obtained her PhD from the Computer Vision Lab of ETH Zurich.

Her thesis focused on dense matching from limited supervision and its applications. She is now a research scientist at Google.

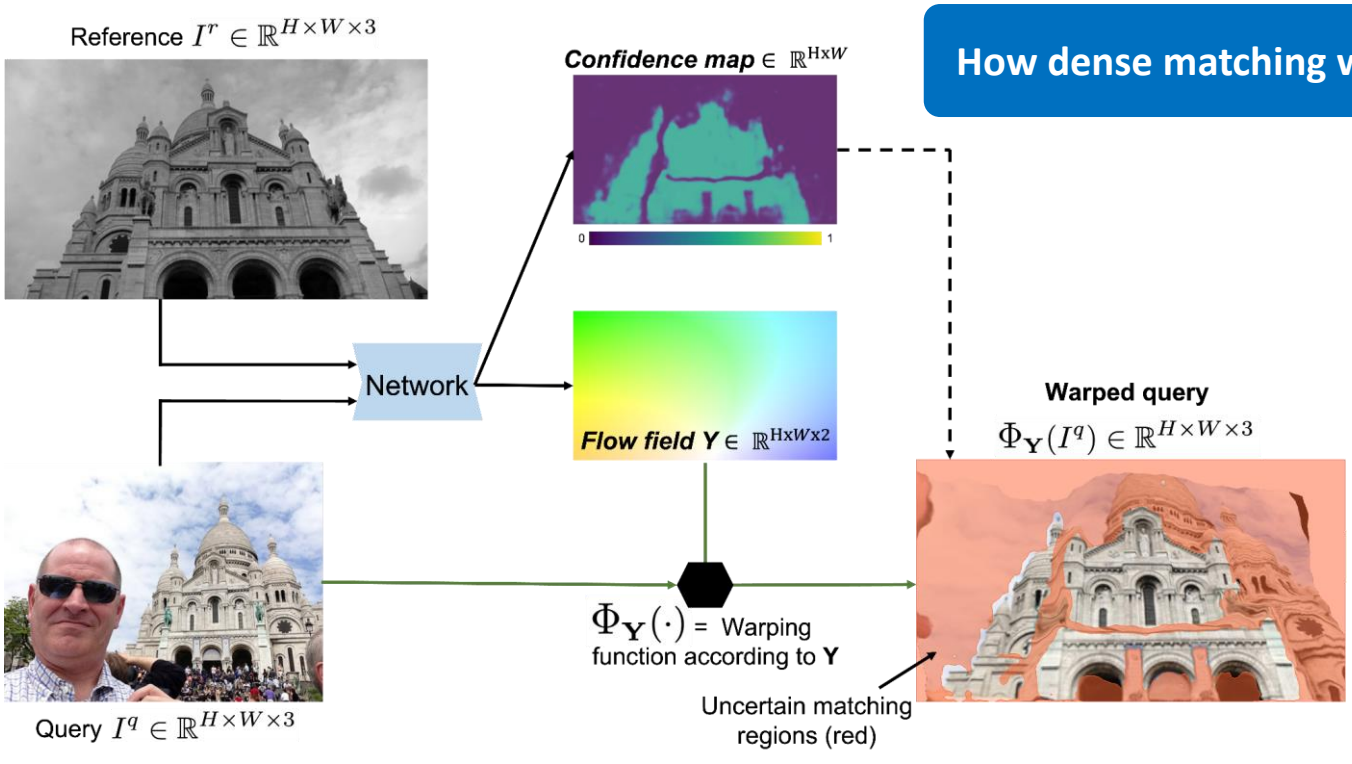
Congrats, Doctor Prune!

Establishing robust and accurate **correspondences between a pair of images** is a long-standing computer vision problem with numerous applications, such as structure-from-motion, image registration, or image manipulation.

While classically dominated by sparse methods which find matches for salient keypoints, **emerging dense approaches offer a compelling alternative paradigm**. Given a pair of images, the goal of dense matching methods is to predict a match for every pixel within the images. It is commonly achieved by predicting the flow field, encapsulating relative displacements relating one image to the other.

Compared to sparse approaches, **dense methods avoid the keypoint detection step**, which is the main failure point of sparse approaches. Moreover, instead of solely relying on descriptor similarities to establish matches, they additionally learn to leverage, e.g. local motion patterns and smoothness priors.

How dense matching works



Despite these advantages, when I started my PhD, dense matching methods had almost only been explored for optical flow, focusing on consecutive views of a video. In contrast, the **more general dense correspondence problem under large appearance and viewpoint changes** had received much less attention.

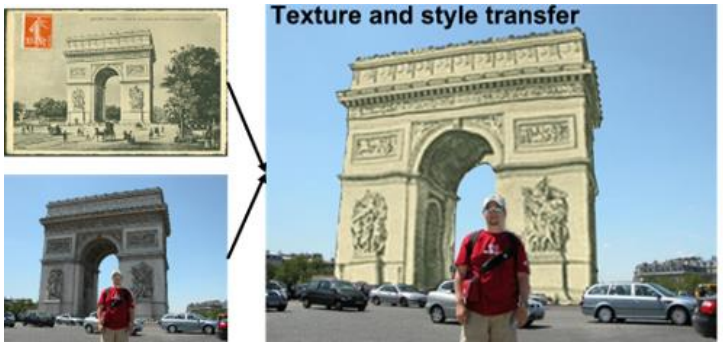
Thus, I focused on this **general dense correspondence problem** in my PhD. In particular, I tackled three main research questions, which I perceived to be the main bottlenecks in dense matching.

What architecture is suitable for dense matching? Most existing dense correspondence architectures were specialized for small appearance changes and limited displacements. I proposed novel architectures capable of handling arbitrary large viewpoint and illumination changes, while still producing sub-pixel accurate predictions. I also introduced an online optimization-based matching module, to improve the network's robustness to repetitive structures (such as windows) or low-textured areas (such as walls).

How to train such a network? Obtaining dense correspondence ground truths for real-world image pairs is extremely challenging, if not impossible. To address this, I proposed two unsupervised training frameworks to train dense matching networks from single images or pairs of images, without any additional annotations. This enables large-scale training on real-world images, while also providing an easy way of customizing a model for new domains.

How to select the "good" matches? Dense matching methods predict matches for every pixel, even in areas that are occluded or for which a match is ill-defined, such as in the sky. This greatly limits the usability of dense matching in downstream tasks like 3D reconstruction, which need highly accurate matches as input. I proposed a probabilistic formulation of the flow prediction, which pairs the matches with a confidence map, reflecting their accuracy and reliability. This confidence prediction enables the direct use of dense matching approaches in popular applications such as image-based localization or style transfer, by filtering out unreliable matches.

Our code is open source on **GitHub at PruneTruong/DenseMatching**. I hope my thesis will inspire others towards the wonderful world of dense correspondences.



Some applications of my work on dense matching

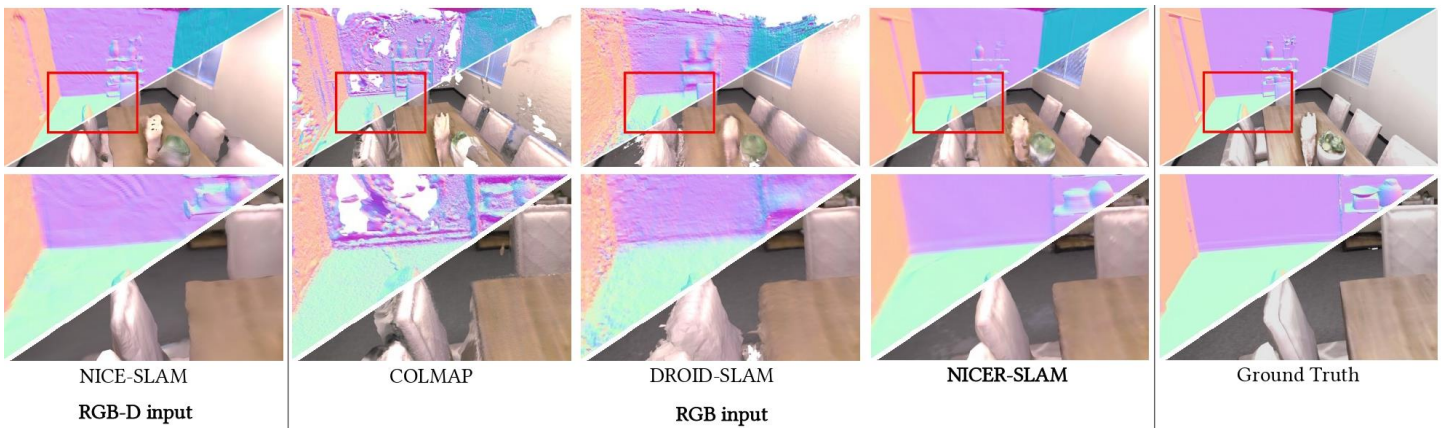
NICER-SLAM: Neural Implicit Scene Encoding for RGB SLAM

Songyou Peng (left) is a Senior Researcher/Postdoc, and Zihan Zhu (right) is a Direct Doctorate student at ETH Zurich. Fresh from winning a Best Paper Honourable Mention award at 3DV 2024, they speak to us about NICER-SLAM, an extension of their previous work, NICE-SLAM, which is looking to overcome the limitations of traditional SLAM systems.

Classic SLAM systems focus on accurate camera tracking results but **often struggle with accurate mapping due to their reliance on sparse point clouds**. By incorporating **Neural Radiance Fields (NeRF)**, Songyou and Zihan aim to achieve precise camera tracking combined with robust surface reconstruction and enhanced color modeling for applications such as novel view synthesis.

This NeRF-based approach, however, would usually depend heavily on depth sensors, which are not as universally accessible or usable in any scenario as RGB sensors. Depth technologies like **Microsoft's Kinect and Intel RealSense** are expensive, and they can face challenges when capturing information in certain lighting conditions during outdoor scenes.

NICER-SLAM represents an extension of the pair's previous influential work, **NICE-SLAM**, but with a significant twist – it removes the depth sensor. This shift to monocular or **RGB SLAM** (represented by the extra 'R') opens up the technology to a broader audience, including those



3D Dense Reconstruction and Rendering from Different SLAM Systems. On the Replica dataset, the authors compare to dense RGB-D SLAM method NICE-SLAM, and monocular SLAM approaches COLMAP, DROID-SLAM, and their proposed NICER-SLAM.

with devices lacking dedicated depth sensors, such as smartphones. "*Not every phone has a depth sensor, but nearly every phone has an RGB sensor,*" Zihan explains. "*We want to make it more suitable for everyone to use.*"

Removing the depth sensor presents a significant challenge, as it plays a crucial role. "*Making it RGB only is not as simple as just changing the name,*" Songyou points out. "*We experimented a lot at the very beginning. We tried many*

things that did not work out and realized the depth sensor was super important for initialization and resampling."

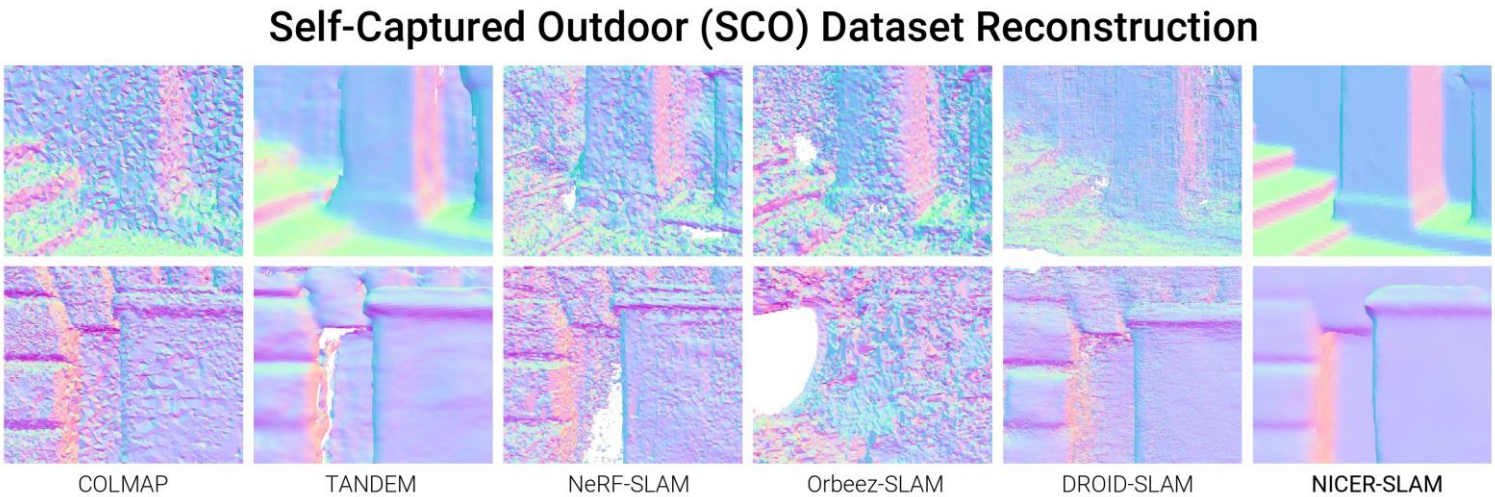
"We show really good results in novel view synthesis and surface rendering and decent results on tracking as well!"

The breakthrough came when **integrating monocular depth and normal priors**, enabling the system to complete extensive sequences and significantly improving over previous attempts that often failed early in the process. "*We always failed in the first 200-400 frames, which was really frustrating,*" Zihan recalls. "*As soon as we tried this monocular and normal depth, it improved, and you can at least finish the whole 2,000-frame sequence.*"

Further enhancements were achieved by **incorporating warping loss and optical flow loss**, bolstering the system's accuracy.

Following a well-received presentation at **3DV** by Zihan, the team took home a **Best Paper Honourable Mention** award for their efforts, a recognition that caught both authors by surprise and one that came at the end of a long road that began with **rejections from SIGGRAPH and SIGGRAPH Asia last year**. "*Most of the reviews were nice, but they didn't think SLAM was suitable for the SIGGRAPH community,*" Zihan reveals.

What do they think the judges at 3DV saw in the work? "*If I try to think about why we got this award, I'd say it's because we have good visualizations,*" Songyou asserts confidently. "*We show **really good results in novel view synthesis and surface rendering** and decent results on tracking as well. Also, we provide a solution for a challenging task. It might not be the fastest, but we provide a clue that it's possible.*"



Zihan was a third-year bachelor's student when NICE-SLAM was released, and he started working on NICER-SLAM while he was in his final year. In addition to making an impressive duo, Songyou and Zihan have worked with renowned researchers **Andreas Geiger** and **Marc Pollefeys** for several years. "*They're both very professional,*" Zihan tells us. "*They know a lot. They can think of papers from 20 years ago that directly relate to what they're doing now. We presented something quite different from what they did during their PhD, but they understood it and could quickly point out their questions, thoughts, and possible improvements.*"

Songyou agrees: "*They're the smartest people I know. Andreas is super sharp and super organized. He gives good insights about your questions and how to solve something. He's a very nice person as well and always encourages us. Marc is super smart and great at geometry or SLAM-related stuff, like structure from motion. He gives great feedback and provides a relaxed environment in which to explore different things under the big umbrella of 3D vision. He gives you the freedom to do whatever you want. That's one of the reasons this work was possible.*"

NICER-SLAM arrives at a time when the field is rapidly advancing. Technologies such as **Gaussian Splatting** mark the next generation of NeRF, and many papers are trying to make it work in a similar SLAM setting. Yet, even in this fast-evolving landscape, NICER-SLAM stands out for its innovative approach to making SLAM more accessible and versatile, demonstrating robust and competitive performance against recent RGB-D systems.

Katherine Scott is a Developer Advocate at Intrinsic, working on developer advocacy for ROS, Gazebo, Open-RMF, and open-source software in general. She is here to tell us more about her career to date and what she is working on now.

Kat, what is your work about?

We have this large open-source project, and my role there is to help people be successful with that project and make people aware of it. Open source – people have varying opinions on it. I see a lot of people being successful with it. My role is to basically help people be successful with open source in building robotics.

What is your opinion about open source?

The way I've phrased this before is that there are two types of companies in the world: there are people who use open source to solve problems, and there are liars because I've never worked in an organization where everything was written truly from scratch. Software exists within a set of frameworks that are written by other people, and most of those people and most of those things end

by Katherine Scott

up being open source or touching open source in some capacity. Just like you can write a book, but all books or all music is sort of derivative work, in some sense, most software is a derivative work. I think it's important that the body of work that we derive our work from is fully expressed, that there are all the things that you would want in it, and that we're at the leading edge of capabilities.

Is this important enough to dedicate some of the best years of your career?

Oh, absolutely. My job has morphed in the past few years. I'm currently working at the job that I said would be the dream job for at least 10 years of my life. I tend to have one rule about working, which is I only do cool stuff, and usually, if I go somewhere, I already know what I'm doing next. This was a thing I wanted to do for a long time. Not only that, one of the roles I wanted to do for a long time. I think particularly robotics is this growing field. If you look at something like, say, the early days of the internet, there were a lot of open-source things, there were also some proprietary things, and the open-source nature of the early

internet is what led to its success. I feel like the same can be said about robotics. We're not going to make a lot of progress, and particularly small organizations aren't going to make a lot of progress unless we have a diverse and rich, open-source ecosystem.

What do you think we need to get there?

Generally, at least with respect to open source, and particularly with respect to ROS, we have an awareness problem. I don't think people understand it exists. I personally think the thing with software is in the early days, you had to write everything, and then somewhere in the '80s and '90s, we kind of said, 'Oh, we should write libraries and make our lives easier and inside of those libraries, we'll put the things we use every day.' ROS ends up being where we put all the things we use for robotics, and you can take from that body of work what you want and leave what you don't want. I always see people having this discussion of it's either open source or no open source, and it's more like we're going to take this set of things from over here, and we're going to build this set of things over here because we have this particular set of requirements. When you have binary thinking where it's like this or the other, it's not generally productive.

The other thing to remember after that is that it's like a national park. You should visit it, you should go do things there, but you should also leave it better than you found it. If you can pick up some trash or clean something up along the way, we'd really appreciate it.

Do you not mind cleaning other people's trash?

Every software job involves a little bit of taking out the trash. [*she laughs*] It's like the young students that come in, and they think that all they're ever going to do is write greenfield code. The reality is you'll be cleaning up people's things forever, and that's part of the job. I feel like most of us should spend a little bit of time leaving the open-source world better than we found it.

My readers are aware of this position because it is similar to Yann LeCun's, whom I interviewed 6 months ago. Do you think it is the

same direction?

Yeah, I'm not familiar with that interview, but I would probably share the belief that open source is the most important thing you can do as a software engineer. The reality, just to expand on that a little bit, I've had these jobs where I've built things that aren't open source. You go and build it, and especially with startups, the likelihood that that organization will be successful is extremely low. The effort, the love, blood, sweat, and tears that you put into the thing most likely will never see the light of day. When we all work together on this shared body of work, you can pick up from these organizations, walk away, and that beautiful thing that you wrote and contributed to gets to stay out there in the world, and you can take that bag of tricks that you just created and bring it to your next organization.

When you started your career in the computer science field, what were you looking for?

I wasn't precisely clear on what I wanted, but I did know that I wanted to only do interesting things. I mostly succeeded at that. If I were in a position where I was writing accounting software or doing financial engineering for some rich person, I wouldn't be happy. I wanted to do research, but more towards the actual - let's get the research out in the field, do actual things, and actual interesting things. I've done a lot of different weird things, and that's kind of my dream, which is interesting.

If I have a microphone for a second, one thing I see that I really kind of dislike is that you get a lot of young students, and they're like, 'I am going to be a robotics perception engineer, and that is precisely what I am going to be, and what is the career path to get me there?' It's like, look, I'm a Developer Advocate right now, but I've worked on satellite data, I've worked in robotics, I've worked in a mapping vehicle, I've worked on microscopes, and, yeah, it generally is computer vision and hardware stuff, but it's never like, 'I am going to be precisely this, and I am only going to be doing this kind of job and I am only going to look for this kind of job.' I don't think I ever said that. It was always like, what's interesting?

To be honest, if I look and say, what do I want to do next? All the interesting things might be completely not robotics. [*she laughs*] Like some stuff I'm really interested

in right now outside of work, maybe I could go do that? I'm in love, just absolutely in love with iNaturalist right now. I'm really interested in what happens when you get a bunch of really smart people about biology collecting a bunch of data, and what if we gave them sensors? That's kind of the cool things that I think about when I'm not at work lately. I think it's good to have lots of interests. I think there are a lot of personality types that are very motivated by new and interesting problems.

I find this very refreshing because I often interview people who have done a PhD, while you took a different decision. Is this connected to the career choices you were talking about?

I wanted to get a PhD when I was younger. I think this is an important story to hear. I might anger a few people by saying this, but…

Please do.

I spent five years in undergrad. I got two degrees: electrical engineering and computer engineering. I did a math minor. I worked in research labs, and all I wanted to do when I was 23/24 was, 'Oh, I'm going to be the next biggest, brightest computer vision professor.' I had no money coming out of undergrad. I was $100,000 in debt. I needed to work. I couldn't afford to eat. I don't come from wealthy beginnings. I went and worked for a few years, and by the time it started being like five years, and I had some savings, it was like, I really should go do this. Then I went back to graduate school thinking, 'Yeah, I'm going to do this PhD thing.' I actually turned down working with some friends at a couple of startups to go to graduate school.

Now, I will say this: I went to graduate school in New York City, which I think is a different experience for a lot of people. Being in New York City and being in the Maker scene there a little bit changed my mind. I came to this realization that I'm 30, I'm getting my master's degree, and if I were to go and try to do this PhD thing, I'd be 40 before I'd even have a shot at being a junior professor and then it's just doing this one thing indefinitely. The probability of being successful in that is incredibly high. The reality is, a lot of times I'm hiring people that have a PhD, but maybe not a PhD in computer vision or something like this because we produce too many PhDs. They all want to do research, but the reality is there aren't that many positions.

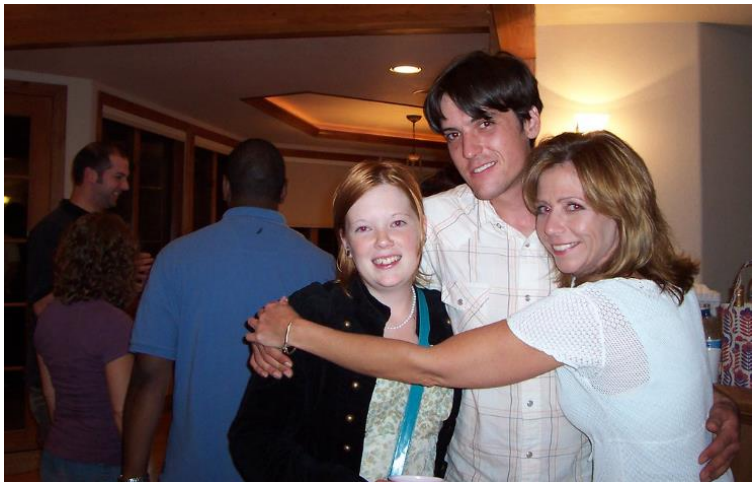
Coming to terms with the fact that you can do really interesting, impactful work, a lot of times taking work that's coming out of a research lab, where we're expanding on that work and bringing it out to the real world, it's just as good. You don't have the title. Your parents can't say, 'My daughter's a doctor,' or something like that, but my parents can say, 'Look, my daughter worked on all these satellites, my daughter started two companies, my daughter works at a Google subsidiary now.' These are things that are still on the table if you decide you don't want to do a PhD. I think there's a lot of work to still be done about making that kind of education more equitable and accessible to people who don't come from the most affluent of backgrounds. That's my quick take on that.

Do you think you would have been more of an insider in the computer vision community had you pursued a PhD?

Oh, probably, I think, yeah. I guess I'm more motivated by applications than I am motivated by research. It's more of, 'Here is this thing that we think we can make in the world, that I think that we can make exist. Is there research that we can leverage to get from where we are to where we want to be?' That's a little bit different from, say, a PhD who's like, 'Here's the state of the art. We're going to push the state of the art just a little bit further over the line. We're just going to get a little bit higher on this metric.' That's good work, and sometimes I think there's really innovative work, but I'm more satisfied by saying, 'Hey, we did this thing out in the real world.' Doing that, I will say, is really difficult with computer vision. If you look at it, there really aren't that many things that are out there in the world. I mean, there are a lot, but by and large, it's very difficult to do.

How do you see this science evolving in the next decade?

I have a slightly contrarian view. I do think that we'll continue pushing the edge of deep learning, but I suspect, I could be wrong, that some of these deep methods will eventually reach some plateau level. This has been the case with most research since we've done research. We have some innovation, that innovation comes to its logical conclusion, and then we



take another big step. At some point, there is going to be a big step. That is not to say that there isn't a ton of interesting work left. I want to see so much more work on things like NeRFs and speeding up the 3D reconstruction and recognition process. There are tons of things that I think we can solve with the existing technology, but I also think that there are interesting things that are the next step beyond that.

One thing that really bothers me a lot is we kind of have decided that more or less 3-channel RGB images are all there are in the world. I've seen stuff coming up, microscopes and satellites, and realizing that, hey, you can do different parts of the spectrum, you can do different bit depths of images, you can do crazy things like event cameras. There are different ways of tackling a similar problem that is how nature seems to solve a lot of problems. I would imagine that that's important to solving problems in the real world. You look at all these animals that have different perceptual systems, they probably have a different perceptual system for a really good reason to solve their particular problem. The thing that I'm really interested in is tackling that little bit of it.

I went to SPIE Photonics West right before the pandemic, and you look at the number of hyperspectral sensing technologies that are commercially available, and they're starting to really ramp up. I think that's going to be an area where there's a lot more interesting things. If you take an interesting problem like, say, plastic sorting or determining if a plant is healthy or not because you're doing some sort of robotic agriculture, it may very likely be the case that these hyperspectral systems take a problem that, you know, you're basically stacking up like five GPUs to do the processing with your conventional camera, or you can just do this one weird trick where you use a different camera and the problem turns into something that can be trivially solved with a threshold or something. That, to me, as an engineer, seems like a better solution. [*she laughs*]

I see a wonderful laboratory behind you. Is there anything there you would like to show us in more depth?

That's my partner's workbench. We split a lab. I don't have anything super cool in front of me right now. I do a lot of boring manager work – I don't get all the cool toys!

What do you dream of achieving before the end of your career?

I have this secret dream that one of

these days I will hit the startup lotto, and I will be able to retire out to some beautiful tropical island or down to Central America and have my nice little house and all day I get to work on a new computer vision – well, I don't want to say new, I want to say it'll derive from most of the things that already exist, but a new computer vision tailored specifically to making it easier to solve computer vision problems for all of the weird types of images that are out there. Not your standard 3-channel RGB 8-bit images. I want a library that makes it really easy to do hyperspectral things, to work with satellite data, to work with microscope data, to work with the kinds of cameras we've never seen before. I've worked with thermal camera data in the past. It's a pain in the butt. The limiter is not the camera exactly; you've got to change some beliefs about it.

That's the thing that I would like to do. I would like to spend my entire career sitting around an island probably spending most of my time writing that and maybe going out and teaching kids on the side. That would be perfect. If I didn't have to worry about money, that is probably what I'd be doing.

To hear more from Katherine, why not check out her very candid (and sarcastic) Twitter/X account?

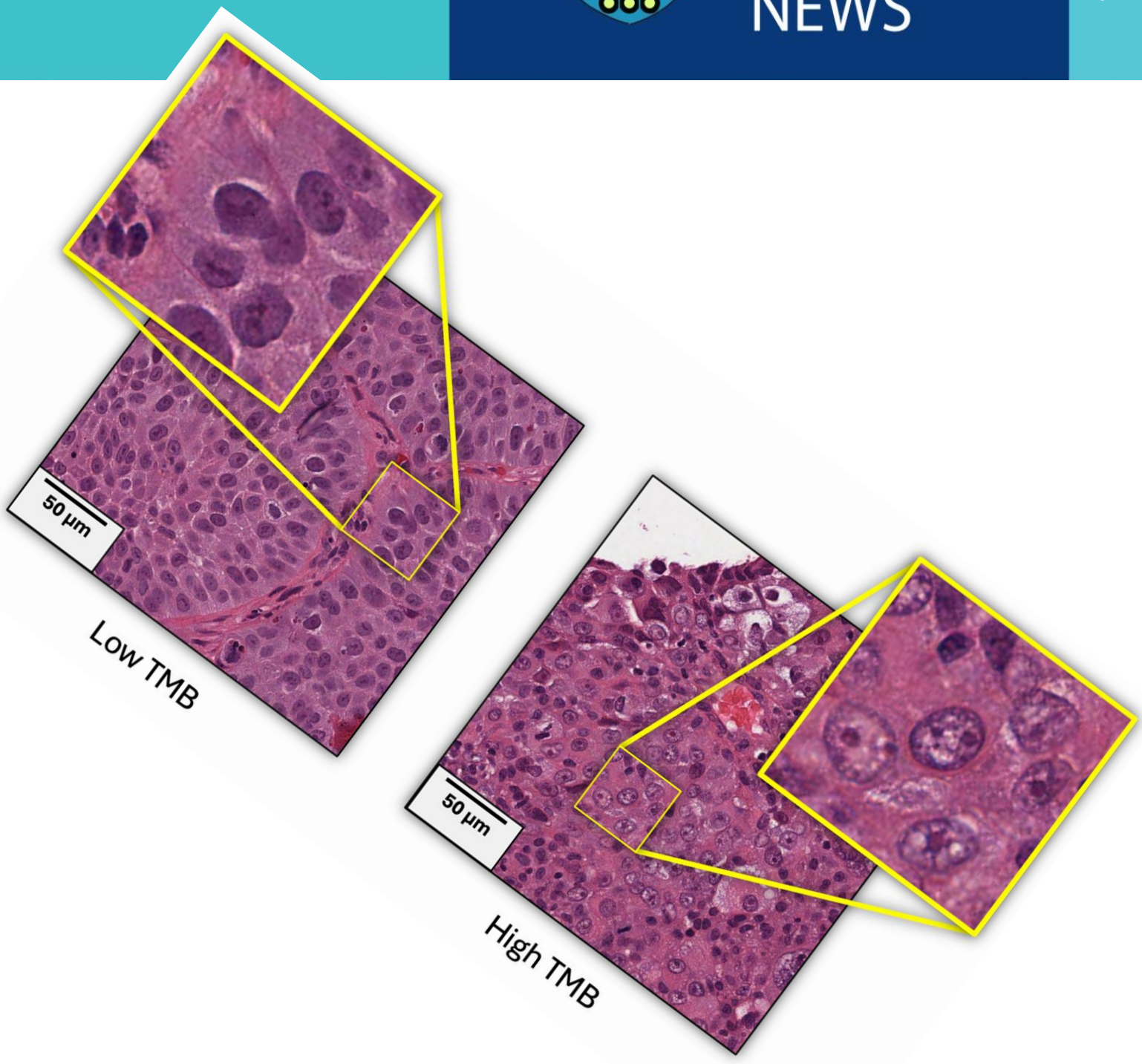
Justus is one of the most outstanding researchers I have ever had the opportunity to work with. His works, such as Face2Face or Deferred Neural Rendering, have fundamentally changed modern computer graphics and how we think about rendering in general, showing us applications that were unimaginable before. I can't imagine anyone else more deserving for the Eurographics Young Researcher Award, and I look forward to seeing more outstandings work from his new group in Darmstadt!

Matthias Niessner

Computer Vision News first published Justus' work just more than eight years ago, when we reviewed his mythical paper Face2Face. That's the same paper that two months later made sensation at CVPR 2016 in Las Vegas, with that famous live demo by Matthias. It was obvious that this was exceptional work!

Ralph Anzarouth

MEDICAL IMAGING NEWS
MAY 2024



An example illustrating potential morphological differences between lung squamous cell carcinoma tissue with high and low tumour mutational burdens. Note the differences in cell organization and the staining of the nuclei.
MORE ON PAGES 52-53

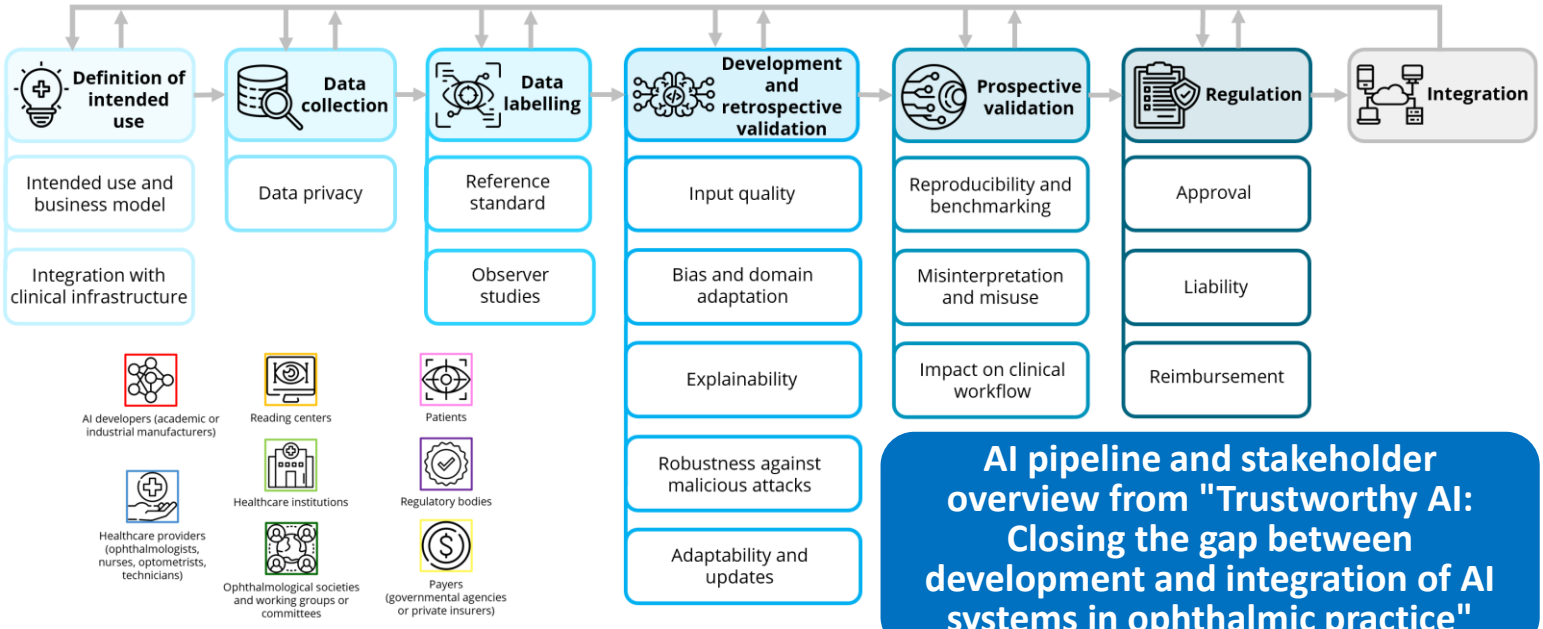
Trustworthy AI in ophthalmology

by Christina Bornberg
@datascEYEnce

It's time for another deep learning in ophthalmology story! I am Christina and my goal is to highlight researchers in the field as part of the datascEYEnce column! This time, I interviewed my (almost) name twin Cristina, who just finished her PhD in trustworthy AI!!

featuring Cristina González-Gonzalo

Cristina is a computer vision specialist and biomedical engineer. She did her bachelor's and master's degree in Madrid before moving on to a PhD under the supervision of Prof. Clarisa Sanchez and Prof. Bram van Ginneken between Radboud University Medical Center and the University of Amsterdam. Her doctoral thesis **'Trustworthy AI for automated screening of retinal diseases'** covers topics such as the reliability of available retinal screening software, attribution methods for explainability, adversarial attacks, and the roles and responsibilities of stakeholders in trustworthy AI.



AI pipeline and stakeholder overview from "Trustworthy AI: Closing the gap between development and integration of AI systems in ophthalmic practice"

The one word Cristina probably used most in the interview was “limitations”: her research sheds light on the limitations of data and algorithms, as well as the requirements and mistrust coming from different stakeholders. And in order to evade those limitations, someone first needs to identify them, analyse them, find possible solutions, and document them - and this is what Cristina did.

Let’s start with the data. Cristina has worked with a variety of medical data and emphasized the importance of understanding the data you are working with. Somewhat, computer vision techniques are similar across medical imaging modalities. However, there remain considerable differences among modalities, for example, when it comes to vulnerability to adversarial attacks (Bortsova, González-Gonzalo, Wetstein et al., 2021). Different imaging modalities have different limitations and this obviously shall influence the choice of the most suitable algorithms and techniques.

In order to find a suitable algorithm, we need to figure out what makes an algorithm suitable for whom. The first step is to start the conversation with relevant stakeholders, such as the clinicians (or healthcare providers) who are meant to use the algorithm. Cristina told me that you have to take notes on variables that clinicians think are important, for instance, the severity of a disease or specific phenotype information. The next step is data exploration and figuring out which data to consider in the analysis. It is indispensable to understand demographics and identify biases that might be present in the data. In the development phase, we shall address potential biases by adequately pre-processing the data and/or adding constraints for bias

reduction during training.

Now it's time for the analysis of the trustworthiness of an algorithm. But how can we define "trustworthiness"? Cristina identified different properties that a trustworthy algorithm should have, including reliability, explainability, and robustness. Hence, we are not meant to only evaluate performance with a Dice score or a ROC-AUC curve, but also to perform observer studies, provide meaningful explainability, and ensure robustness against potential adversarial attacks.

Let's get into more detail.

Benchmarking for the research community and healthcare providers

Probably the most widely spread approach to compare an algorithm with existing literature is by using public datasets and common computer vision metrics. However, this comes along with the limitation of potentially being irrelevant: public datasets may not be representative of a specific clinical setting and certain populations, and metrics that are common for the computer vision community may not be intuitive for clinicians and patients.

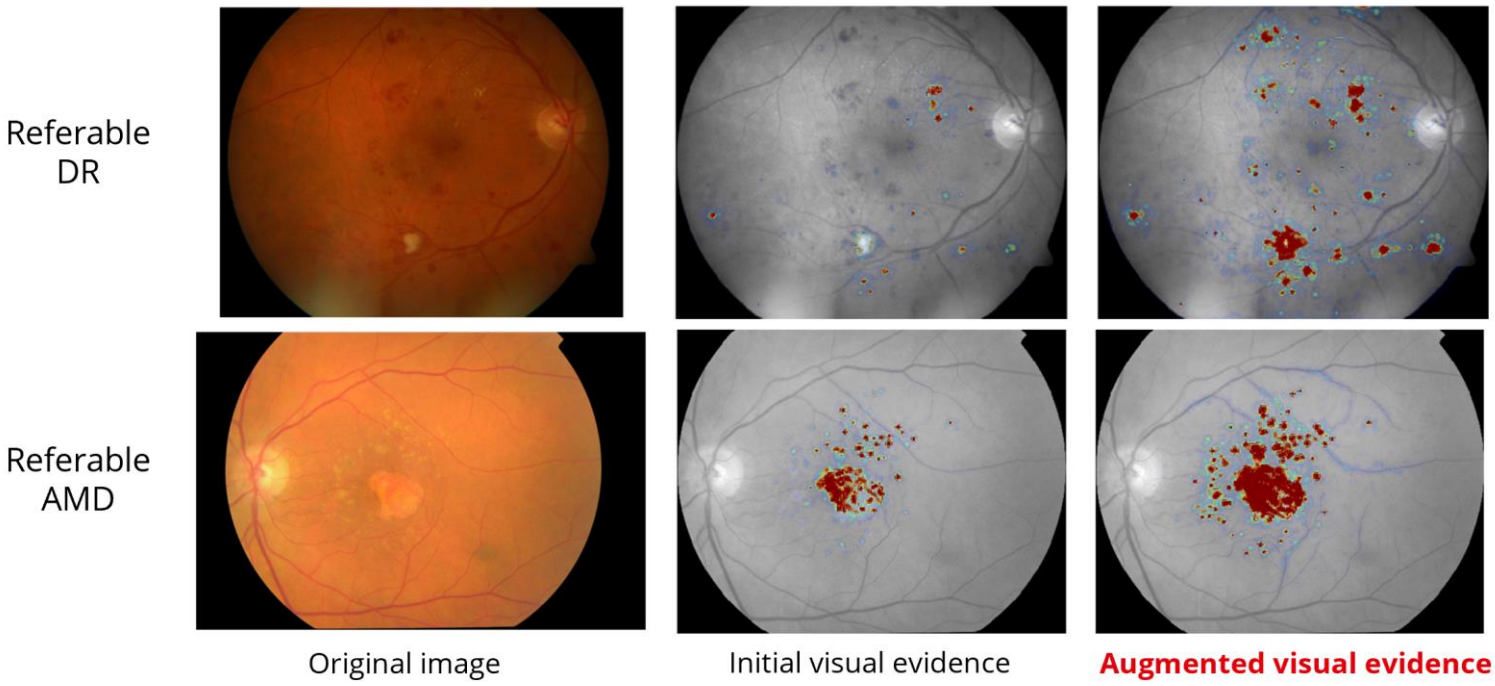
Cristina highlights the importance of establishing observer studies to overcome these limitations. Observer studies allow us to compare clinicians with algorithms in the same setting as well as to analyse inter-reader variability. A potential demographic bias shall also be considered by including specific population groups in the studies. In the context of automated screening of diabetic retinopathy and age-related macular degeneration, Cristina was able to define fair and realistic expectations for commercially available algorithm performance, ensuring they are aligned with what is currently achievable by humans (González-Gonzalo et al., 2019).

Visual evidence augmentation for clinicians and patients

Visual attribution methods are widely adopted in classification tasks in medical imaging, however, Cristina demonstrated that basic heatmaps are not enough to provide meaningful algorithm explainability. They were accompanied by a lot of questions from the clinicians and turned out to be counterproductive and misleading in some cases. Cristina proposed a method for visual evidence augmentation, combining visual attribution and selective inpainting to iteratively uncover abnormalities. Her method allowed her to leverage the "knowledge" contained in an algorithm and generate more exhaustive explanations (González-Gonzalo, 2020).

Adversarial attacks and robustness analysis

Another important aspect concerning the trustworthiness of an algorithm is its robustness against malicious attacks, including adversarial attacks.



Visual evidence results from Cristina's publication "*Iterative Augmentation of Visual Evidence for Weakly-Supervised Lesion Localization in Deep Interpretability Frameworks: Application to Color Fundus Images*".

Susceptibility to such attacks is intricately linked to financial and fraudulent incentives, alongside technical vulnerabilities within the clinical infrastructure. Cristina and her peers investigated the vulnerability of algorithms to adversarial attacks in three different medical imaging modalities (radiology, ophthalmology, and histopathology). Their study showed the effect of common algorithm design choices (e.g., pre-training on ImageNet) on adversarial robustness and the importance of establishing realistic and standardized robustness studies (Bortsova, González-Gonzalo, Wetstein et al, 2021).

Multi-stakeholder collaborations for seamless algorithm integration

After testing the mentioned (and other) properties for trustworthiness, the next step is the integration of algorithms in the clinical infrastructure. In her last study (González-Gonzalo et al., 2022), Cristina highlights how a close collaboration between clinicians, AI engineers, and other relevant stakeholders is crucial to generating algorithms that are properly used and trusted. It is also essential that target users get adequately trained on how to use the algorithms in-house. Clinical validations, ideally prospectively, can be performed for this purpose. So let's hope to see many more of those collaborations in the future!

I want to thank Cristina again for giving her insights on trustworthy AI, congratulate her on finishing her PhD and wish her the best of luck for her future!

RSIP Vision's Dekel Shapira and Artium Dashuta talk to us about their work with Blender, a free and open-source 3D creation suite, to generate synthetic video data that can be used for various Machine Learning projects.

Dekel and Artium chose the **3D content creation software Blender** when a recent project required the creation of **realistic synthetic videos of surgical procedures**. Other programs can render 3D objects and scenes, but the team needed a tool that could seamlessly integrate 3D assets to generate videos that look like real cases.

By knowing the clinical use-case they generated a basic scene. This scene was passed to a medical expert annotator, who moved the tools authentically before it was rendered in Blender. "*Whenever you train a neural network, you want some ground truth data,*" Dekel tells us. "*You need pictures that are as close as possible to what you see in real cases. We saw many videos of real procedures, identified the stages and created a guide to what should appear in the blender scenes*".

Despite its impressive features, using Blender was not without its challenges. Dekel says it is not the ideal tool for programming. "*It has a Python Interface, but it's not very convenient to debug and not very persistent in the API,*" he reveals. "*Also, it has quite a high learning curve. Once you know it, it's very convenient, but it takes time to learn it because it's very different from other programs. It has its own logic*". The creation of the 3D assets is also non trivial and sometimes can required the assistance of a 3D artist. This is worthwhile, since sometimes the rendered results can look so realistic that even an expert will find it challenging to discriminate between a synthetic image and a real one.

Various projects can utilize Blender and its realistic data generation capabilities. Segmentation is probably one of the most straight forward examples. You know where blender placed an object and the locations of rendered pixel for it is the segmentation. However, the benefit of synthetic data becomes much more obvious when dealing with tasks where it is very difficult to obtain accurate ground truth. For instance, soft tissue tracking is very difficult to annotate. When using synthetic data, you have a model that is warped and you control its motion. This means that you have all the information to understand the movement of each pixel.

Artium and Dekel conclude saying that it is not always sufficient to use only synthetic data as intricate details may be missed when using simulations. The best approach is to combine at least a limited real dataset with synthetic data. The more realistic the synthetic data, fewer real cases can be used. This fits well in clinical projects where large amount of real cases are often difficult to obtain.

Dekel Shapira

Data Curation & Augmentation in Medical Imaging

Shuoqi Chen (left) is a medical imaging and computer vision engineer at Intuitive Surgical. Dominik Rivoir (right) is a PhD student at the National Center for Tumor Diseases in Germany under the supervision of Stefanie Speidel. They are here to speak to us about an exciting new workshop they are co-organizing at CVPR next month.

The Data Curation and Augmentation in Medical Imaging (DCA in MI) workshop at CVPR in June aims to attract individuals passionate about **medical imaging and medical computer vision** and eager to explore innovative solutions to the challenges in these fields. Borne out of the organizers' previous experiences at the conference, where they felt a void in terms of medical imaging content, the event creates a platform for the field to intersect with the wider computer vision community, foster collaboration, and reveal the challenges faced in medical imaging to a broader audience.

"Medical imaging is already a pretty big field," Shuoqi tells us. *"We have MICCAI and other conferences supporting that area of expertise, but at CVPR, one of the leading conferences in computer vision and pattern recognition, it's still lacking. We thought, what could we do to expand things further in terms of community building and bringing our folks, who really know the technology and the research, onto a bigger stage with all the experts in computer vision and AI?*

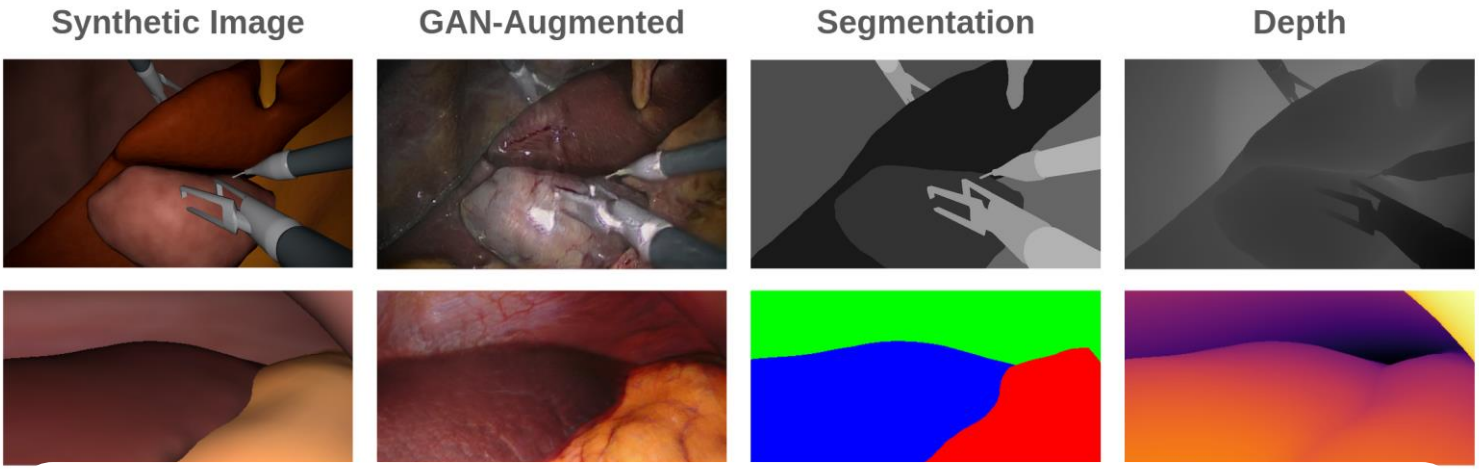
Indeed, the workshop promises to be a place where people from

MICCAI Society Endorsed Event

academia and industry can come together to share a range of viewpoints and perspectives on using data to solve medical imaging problems and, ultimately, **improve patient care and health**. "When I was a grad school student, I always tried to focus on innovation," Shuoqi recalls. "The emphasis was on developing novel models and algorithms to optimize certain metrics. In industry, we're looking to start from the data and develop robust solutions. We have loads and loads of data and want to ensure our method is the simplest and works in as many cases as possible. The contrast between those two fields is what we want to bring to the workshop."

The workshop's agenda is being finalized, but attendees can expect solution-focused content on **data selection, data synthesis, learning with limited and imperfect data, and data verification**, as well as some of the persistent challenges that come with working with data for machine learning, like **data scarcity, annotation costs, domain shift, and bias**. In medical imaging, these issues are compounded by privacy concerns and quality standards. "*In the medical domain, we've been thinking about these challenges a lot,*" Dominik reveals. "*I think this should be a more prominent topic in the main computer vision area. We hope we can interest people in it and get insights in both directions.*"

There will be a range of paper presentations, panels, and keynote speakers. The organizers hope the

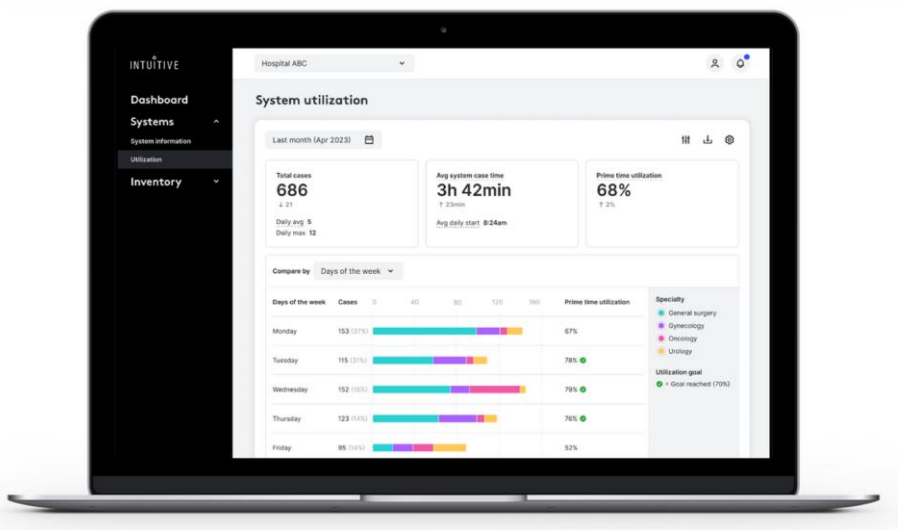


Synthetic data can potentially alleviate issues related to data scarcity and privacy, but it also poses new challenges. From public synthetic datasets for surgery: Top Bottom

main take-home message will be a broader understanding of the data behind medical imaging methods. "*Medical imaging is a really interesting area because we're very close to applications and have many safety and fairness-related issues we have to think about,*" Dominik points out. "*We often think about things that maybe aren't the biggest topics in the main computer vision area, and that's something we want people to think about, have new perspectives on, and maybe use in their future research.*"

At Intuitive, Shuoqi sees the real-world applications of medical imaging and computer vision every day. Through his work on the **robotic lung biopsy tool Ion**, which helps diagnose lung cancer in the peripheral lung area, he knows that precise image segmentation relies on robust solutions that cater to diverse patient profiles and healthcare settings. "*Our goal is to develop **robust and general solutions for lung biopsy**,*" he explains. "*We want a solution that works in most cases – not just one dataset for one hospital. We want it to work for all hospitals.*"

Eagle-eyed readers will know this is not the first time we have featured Dominik in our magazine, and as he tells us, his **November 2021 interview about his ICCV paper** had an unintended consequence that led us to where we are sitting today. "*You're actually the reason Shuoqi and I know each other!*" he exclaims. "*He read your article on my paper on synthetic data and, after trying the method himself, contacted me with some questions,*" he pauses. "*My code's documentation might not have been perfect!*" he laughs. "*We stayed in contact after that and even met at **Intuitive** in California in 2022. Then, when Shuoqi started planning the workshop, with synthetic data being part of the scope, he asked me if I wanted to join the team, which I was more than happy to do. So, thanks again to you – even the first interview is still creating new opportunities!*"



Pinned

Camila González @camgbus · Mar 13

Thank you @BVM_Community for granting my PhD thesis the BVM Award and giving me the opportunity to present it at the @BVM_conf in Erlangen 😊 it is so nice to be back in 🇩🇪 for a visit! 🍻🥨

Congrats to Camila for her award and thank you to Christina Bornberg for the additional photos with Sophia Bano!

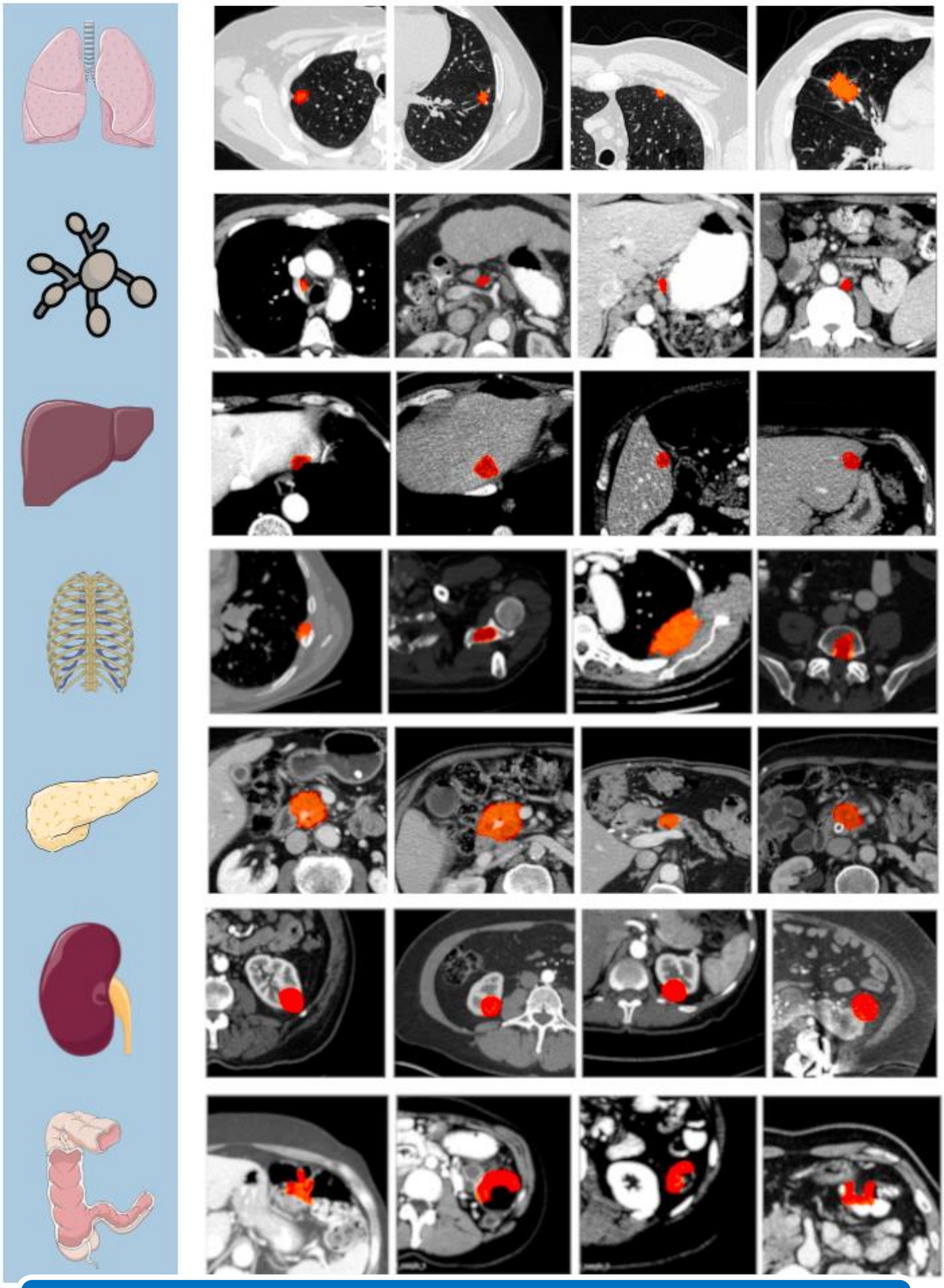
The Universal Lesion Segmentation '23 Challenge

Max de Grauw (left) is a PhD candidate at the Radboud University Medical Center, supervised by Professor of Medical Image Analysis Bram van Ginneken (right). Together with Alessa Hering, they are co-organizers of a lesion segmentation Grand Challenge that has just crossed the finish line. Max and Bram are here to tell us more.

Several successful medical challenges have focused on **AI-based automatic segmentation models for specific tumor types**. However, in clinical practice, radiologists encounter a wide variety of lesions, some more common than others, and a more comprehensive approach to lesion segmentation is needed to handle this diversity.

The Universal Lesion Segmentation '23 Challenge (ULS23) targets the many lesion types in the thorax-abdomen area from the pelvic floor to the neck. "*Our approach here is to pool all the knowledge from these different lesion segmentation challenges,*" Max explains. "*We create a very large and diverse dataset and train a single model to take into account all the ways these lesions share certain morphological features, such as size, shape, and density.*"

The foundation of the challenge is a robust and clinically relevant 3D training dataset curated from a decade's worth of radiological reports from Radboud University Medical Center and Jeroen Bosch Ziekenhuis. The team analyzed the reports to find patients with **standardized lesion measurements** interpreted using the **Response Evaluation Criteria In Solid Tumors (RECIST) guidelines**. They then took a representative sample of those patients and fully annotated their lesions.



Examples of the fully annotated lesion types in the training data

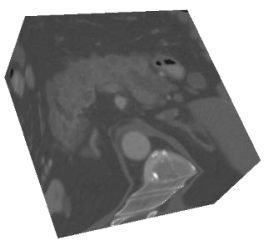
The challenge also uses the 2018 release of the **DeepLesion dataset** by a group of National Institutes of Health researchers led by radiologist Ronald Summers. Although unmatched in size or scope, it was only partially annotated, containing only **the long and short diameters of one slice of a lesion**. The team plugged those gaps by pulling together fully annotated 3D data from publicly available sources. "*Radiologists shouldn't only draw the long axis of a lesion,*" Bram asserts. "*They do that because they don't have time to do a real 3D segmentation. The guidelines should require a proper 3D analysis of a tumor. If you want to assess if lesions are growing in a cancer patient accurately, you shouldn't just draw one line because that's not precise.*"

Despite the diverse training dataset, the team predicts there will still be lesions that are not well represented, for which follow-up challenges will be necessary. "*This is how the Grand Challenge principle works,*" Bram points out. "*If you solve one task, there are always more extensive or new tasks coming up.*"

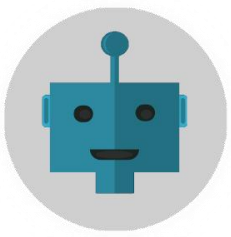
The organizers also collected a varied, multi-center test set, providing a solid evaluation pipeline that will allow models to be benchmarked against the results obtained this year in the future.

Thinking about transformative projects in the broader field of computer vision, Bram says that medical image analysis may now be at the point computer vision was with the ImageNet challenge. "*Before ImageNet, they had PASCAL, with 10 classes,*" he recalls. "*Then ImageNet came along with 1,000 classes – all these different types of cars, airplanes, dogs, birds, etc. We want to do that for medical imaging – to have* ***only one model covering all these different lesions****.*" For Bram, the long-term goal of the challenge is to **develop new tools to analyze lesions in 3D**. He sees it as a collaboration rather than a competition, fostering a community-driven approach to tackling problems, ultimately ending up with a better dataset than you would have if everyone had compiled data in a silo. "*Max developed an algorithm that was the baseline here,*" he reveals. "*We could have*

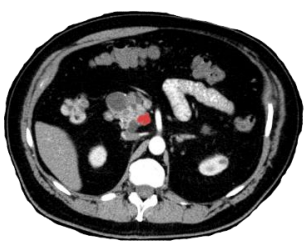
Select Lesion



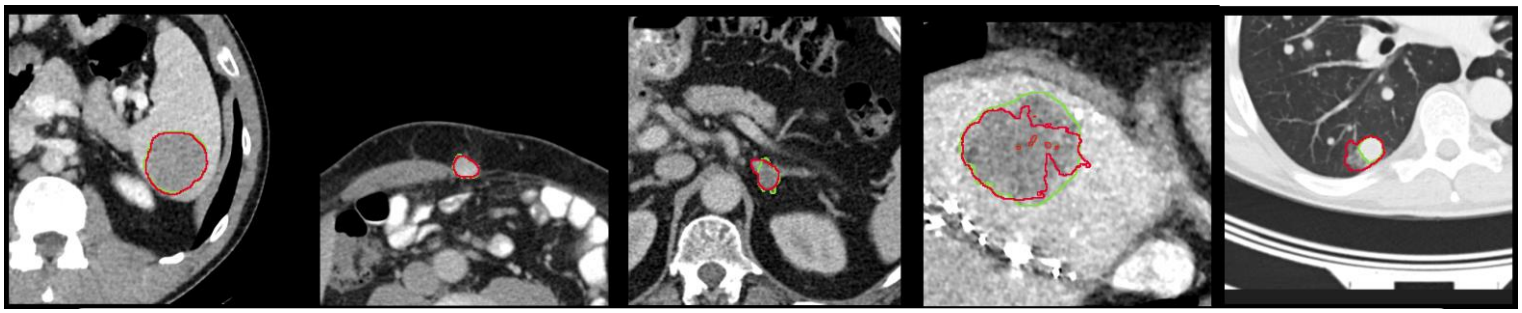
Crop VOI



Run Model



Evaluate



Predictions of the baseline model (red) vs the ground truth (green) on lesions from the test set.

told him to keep working on it and improve it and finish his PhD with one algorithm, but by organizing the challenge, several other groups have made progress now."

As ULS23 drew to a close last month, a little later than anticipated when it was named, seven high-quality submissions showcased diverse architectural approaches and innovative pre-training strategies, with some incorporating additional data from their own centers. "*There's a lot to unpack!*" Max tells us. "*We're really happy with the turnout. The winning solution uses* ***U-Mamba architecture****, for which a paper was released recently, so we've already seen the state-of-the-art applied to our challenge.*"

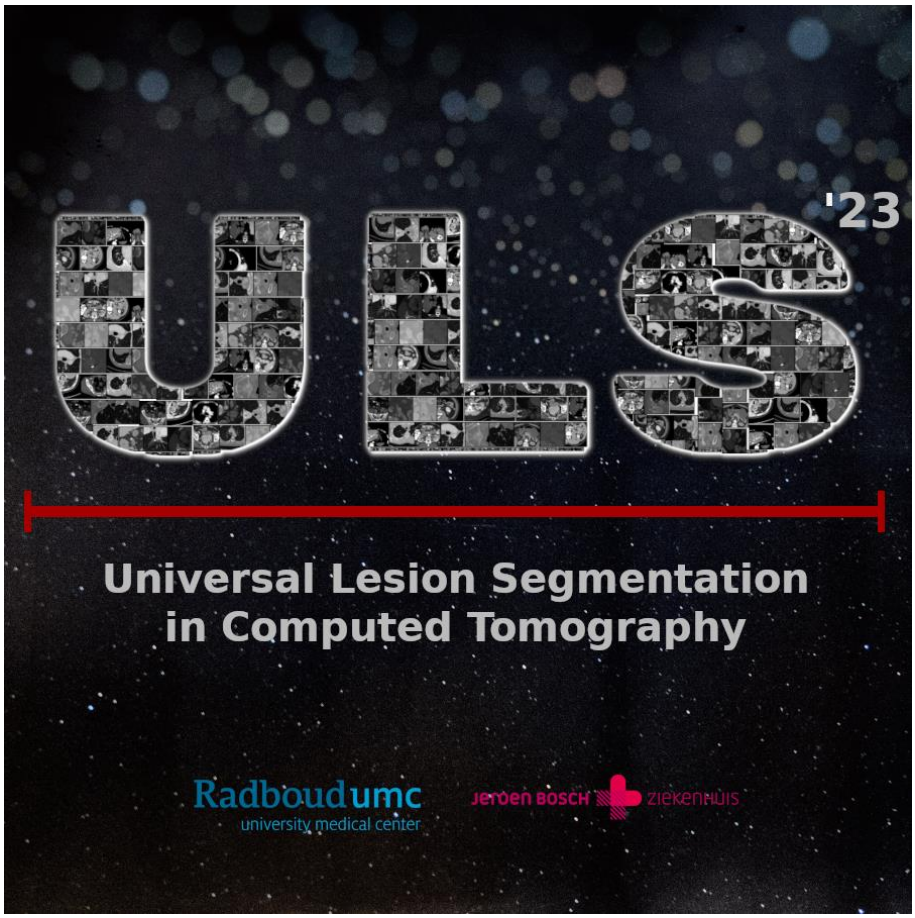
Max and Bram are keen to thank the organizations they've worked with to get the data and the radiologists who read the scans. Bram's group at Radboudumc, the **Diagnostic Image Analysis Group**, has around **60 researchers working on medical image analysis**, with half in radiology and the other half in digital pathology. "*It's quite unique that we have both a big pathology and a big radiology group,*" he says. "*What I think will happen in the next decade is that AI will become prominent in every area of medicine, so we'll probably grow in some other areas, but we'll require new grants and people to do that.*"

In future challenges, they plan to focus on enhancing accuracy. This first edition focused on smaller, faster models, with an inference time that allowed them to be used online so that a radiologist could click on a lesion and have it segmented in under five seconds. "*It'd be interesting to see the upper limit of segmentation accuracy,*" Max ponders. "*To do that, we'd need to remove the time limit and allow for more resources on the platform to run larger models. I think it'll happen naturally over the coming years, but with that will come a need for delving further into what makes a good segmentation.*"

This time, for the test set, they used a single radiologist who 3D-segmented each lesion based on previous annotations from the radiological reports and a team of student annotators. *"It wasn't just a quick segmentation, but to really establish what the acceptable boundary is, you'd need to have* ***multiple radiologists read each lesion****,"* Max identifies. *"That would come with a huge annotation burden, requiring a larger budget and more time, but I think it would push it to the next level."*

Bram's wish for the next edition is for **the computer to segment the lesion and the AI system to show how confident it is that the result is accurate**. *"That's a very important step,"* he affirms. *"If we're ever going to use this in clinical practice, you can't have radiologists check all the computer segmentations because that's too much work. The computer needs to indicate where it's really sure you don't need to check it. Then, the users can determine how much they still need to do manually. That would be a good topic for a follow-up challenge!"*

"The winning solution uses U-Mamba architecture, for which a paper was released recently, so we've already seen the state-of-the-art applied to our challenge!"



Congrats Ruogu and Skylar!

Congrats to awesome Ruogu Fang (top) for receiving the inaugural AI Course Award for her exceptional leadership in developing and teaching the Medical Artificial Intelligence course at the University of Florida. And also to awesome Skylar Stolte (second left) for being awarded the UF Herbert Wertheim College of Engineering Faculty Excellence Award for her groundbreaking work in AI for brain health and dementia prevention. More about their work here.

Salma Dammak (right in the photo) has recently obtained her PhD at Western University in Ontario, Canada under the supervision of Aaron Ward (left) and David Palma. Her PhD research focused on developing AI models to address challenges in the diagnosis and treatment of lung cancer for both pathology and radiology applications.

Salma is now a postdoctoral researcher at the Computational Pathology Group at RadboudUMC in the Netherlands.

Congrats, Doctor Salma!

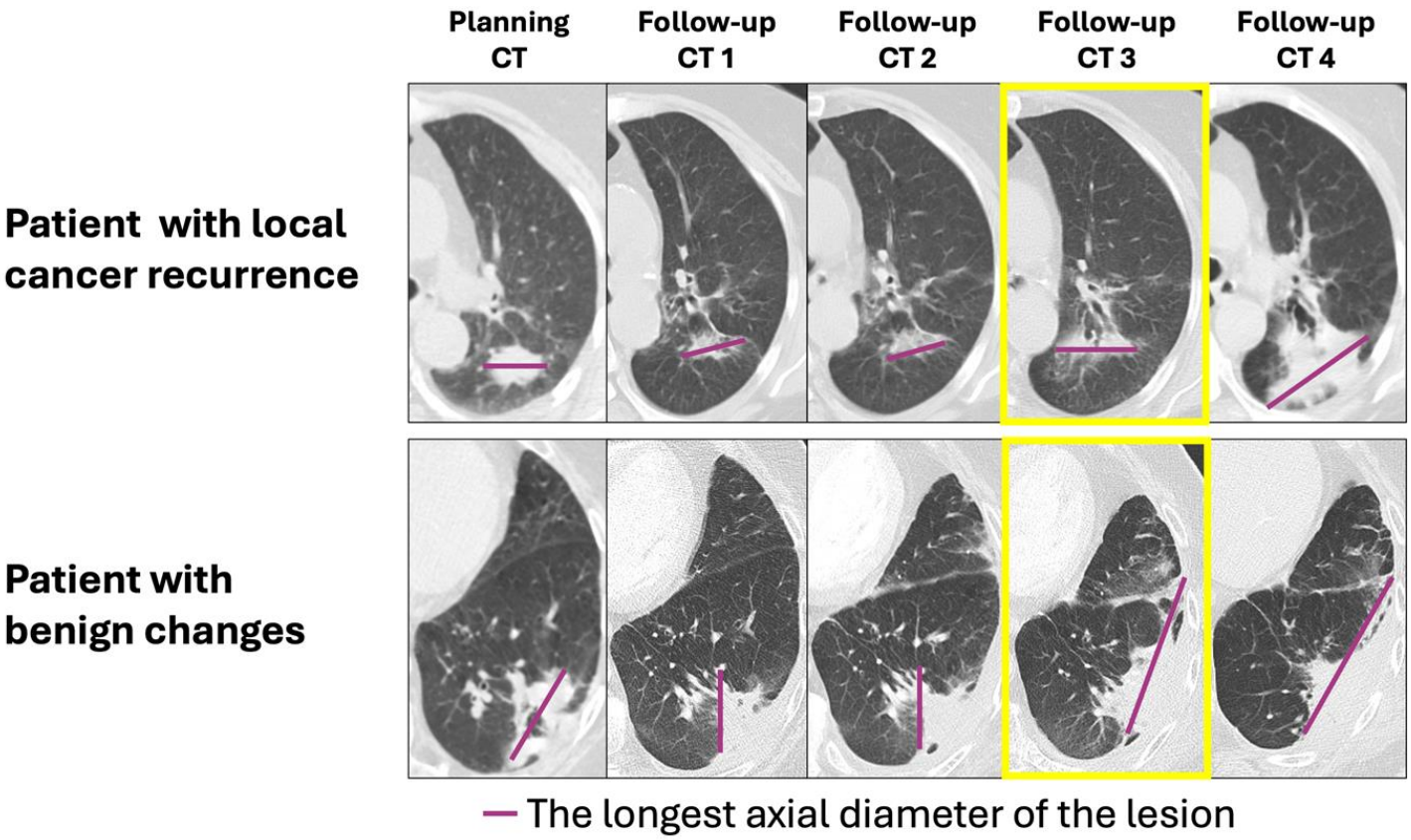
The motivation for Salma's thesis lies in the complexity of lung cancer treatment and the difficulty of achieving fully personalized care given the current tools available in the clinic. For personalized care, extensive patient information is necessary, which may sometimes be incomplete. In her thesis, Salma addresses two clinical challenges where this is the case by using artificial intelligence to build models that predict the missing information.

The first challenge focuses on predicting tumour mutational burden (TMB), a biomarker that measures how mutated a cancer is. This biomarker is important for determining which lung cancer patients are most likely to benefit from immunotherapy, which can be highly effective but carries the risk of severe side effects. Unfortunately, TMB assessment is currently inaccessible in most clinics, as it requires invasive biopsy and costly genetic sequencing. However, this thesis shows that a model can predict TMB from existing lung tissue images based on the morphology of cancer cells on tissue samples taken during previous cancer surgeries, which are common for patients who are considered for immunotherapy. Tumours from surgery contain many non-cancer tissues, and these tissues were excluded manually when developing the model, but to improve clinical translatability, this thesis also presents a model to automate this by identifying cancer cells within surgically removed tumours. This would allow the TMB prediction model to be completely automated.

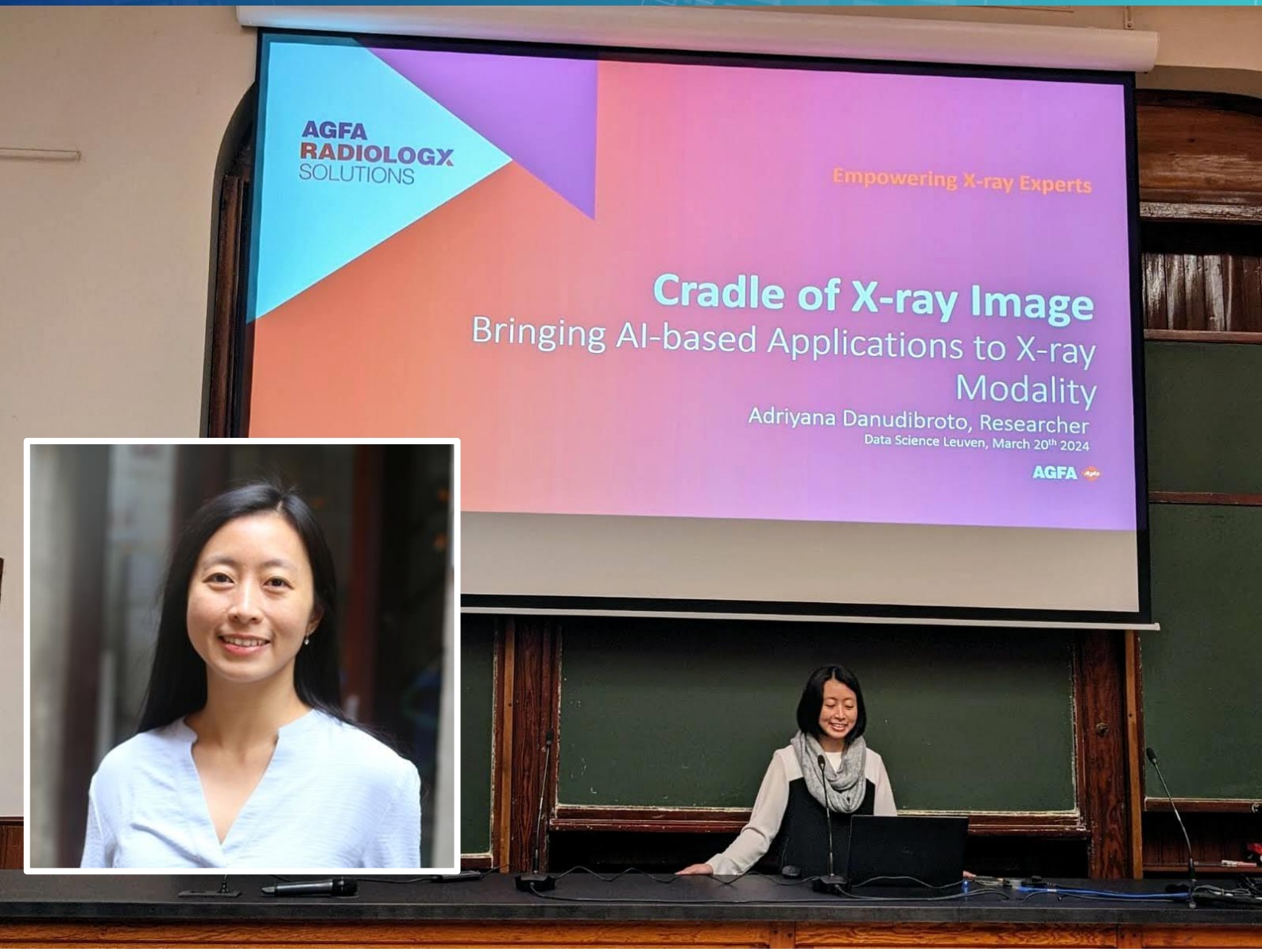
The second challenge involves assessing the success of stereotactic ablative radiotherapy (SABR). SABR is a highly effective and well-tolerated treatment, but it induces benign inflammation and scarring around the tumour, mirroring tumour growth in post-treatment scans. Distinguishing these benign changes from true cancer recurrence is critical, as the latter requires immediate and potentially risky interventions. With current tools, this takes over a year, but this thesis presents a model that can discern the two at the earliest sign of potential growth. The model can do this by analyzing the appearance of lesions on routine follow-up scans, detecting patterns of recurrence before they become visible to the human eye.

These studies highlight the potential of artificial-intelligence-based models to address critical clinical challenges in lung cancer care, particularly in the context of novel treatments like immunotherapy and SABR.

Two of Salma's thesis papers can be found here and here.



Two patients who received SABR, one with cancer recurrence and one with benign changes. Note that for both cases, the lesion appears to grow progressively after treatment, and that further intervention would be used after the third follow up CT scan (yellow box) due to this progressive growth. This may be too late for the patient with cancer recurrence, and would be unnecessary for the patient with benign changes.



Adriyana Danudibroto is a Medical Imaging Researcher at Agfa Radiology Solutions. We asked her to tell us about her recent presentation at the Data Science Leuven meetup.

by Adriyana Danudibroto

As a researcher at Agfa Radiology Solutions, I've been involved in exploring the intersection of technology and healthcare. At a recent Data Science Leuven meetup in Belgium, I had the opportunity to share my insights on integrating AI applications within X-ray modalities. The presentation went over the nuances of AI deployment environments and their implications for radiology.

The choice between processing on the modality, cloud, or edge devices is

more than technical, it's a strategic decision that impacts workflow and patient care. It is important to understand these environments and tailor the AI solutions to operate within their constraints.

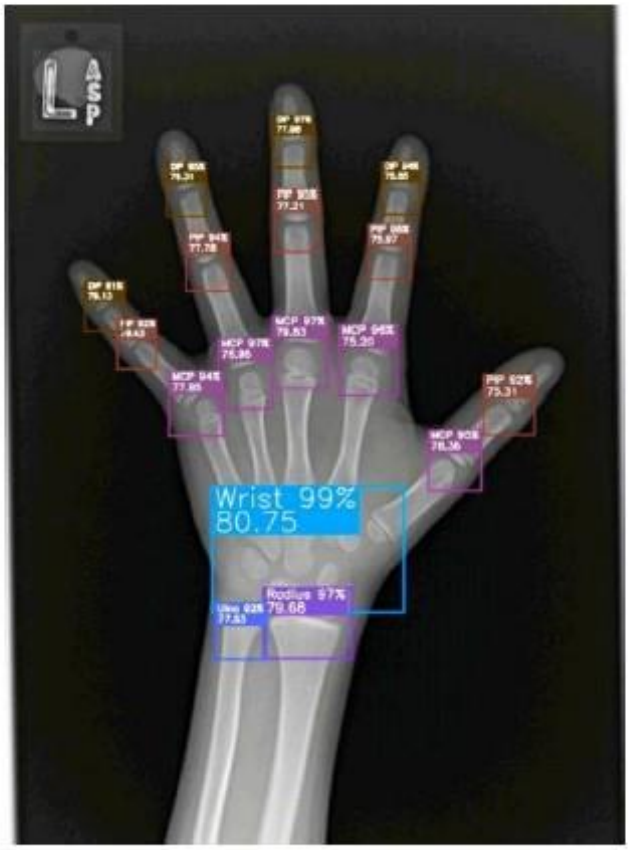
The choice of application needs to be inspired by the real needs of the clinicians.

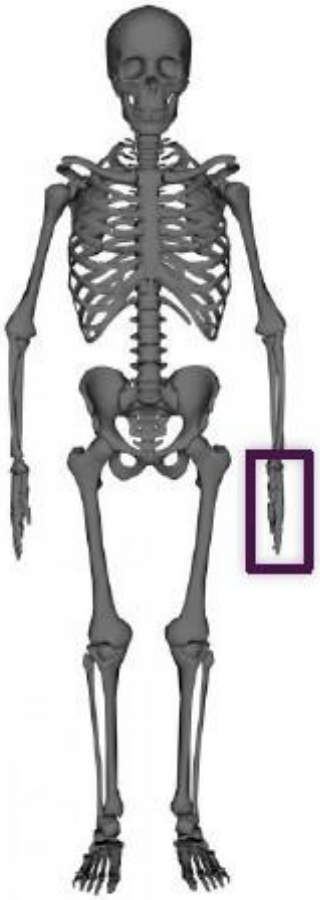
The challenges faced by radiologists, such as the overwhelming number of images requiring diagnosis are our main motivations. We actively investigate how AI can help by improving workflow, aiding technicians, and prioritizing urgent cases.

My key takeaways from the evening are:

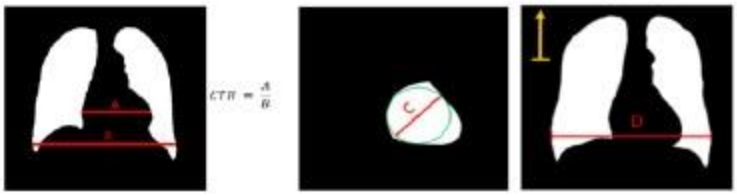
- **Understand Your Deployment Environment**: It's crucial to recognize the limitations of your technology and adapt your AI solutions accordingly.
- **Know Your Stakeholders**: Deep engagement with end-users is essential. By understanding their pain points, we can develop AI applications that address their core needs. Sometimes the simplest solution trumps the fanciest. As an example, our most used AI application is a feature that automatically rotates X-ray image to the correct orientation which based on our case study manages to save 20 hours of technicians'

Bone Age





Cardio-thoracic Ratio



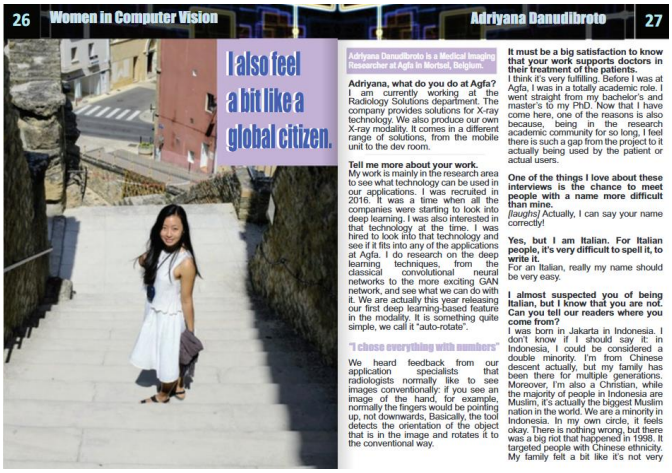
workload for bedside imaging per modality annually.

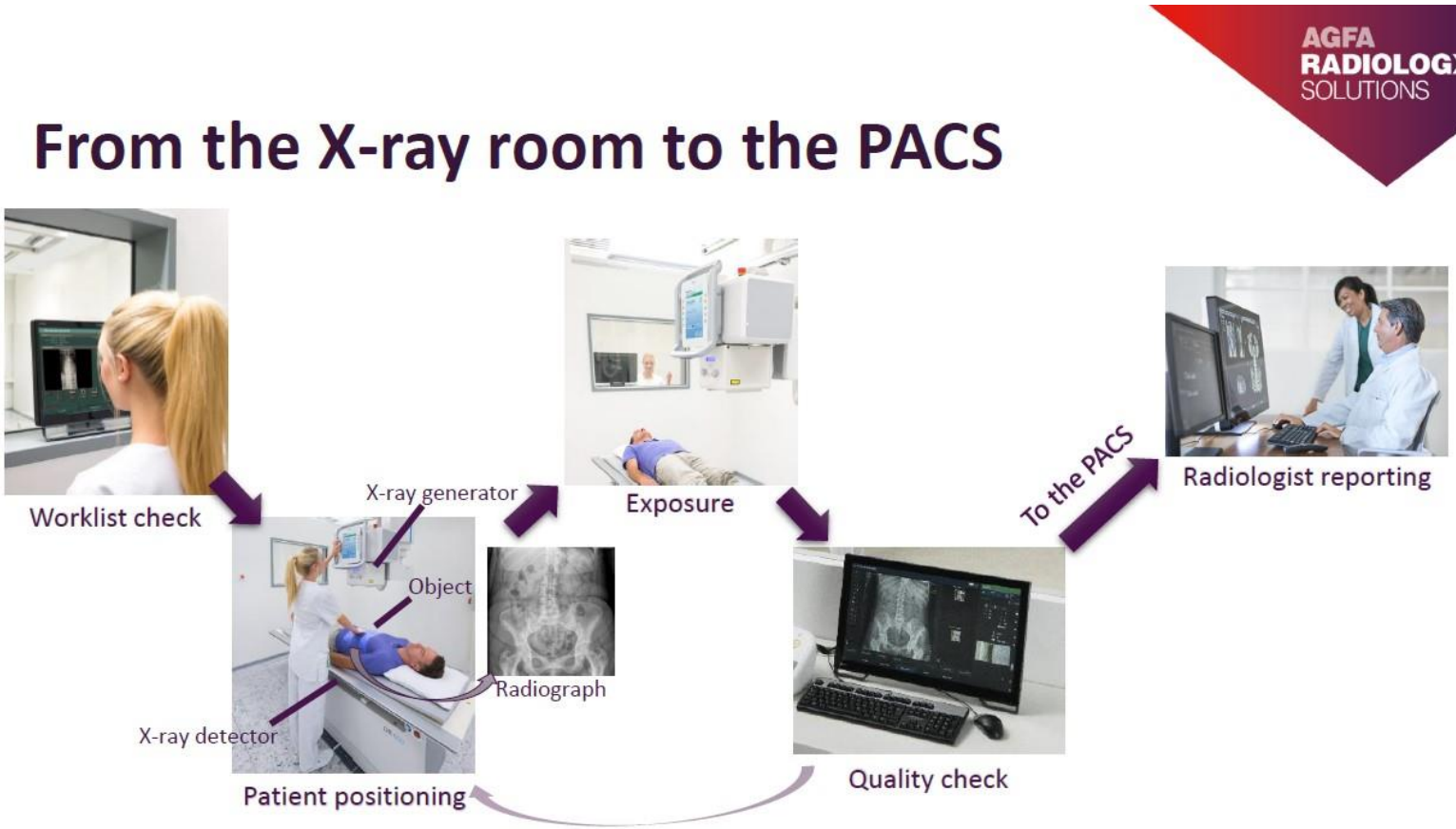
- **Embrace Collaboration**: True innovation often comes from building upon existing foundations instead of starting from scratch. Being open to collaboration can lead to unexpected breakthroughs and advancements.

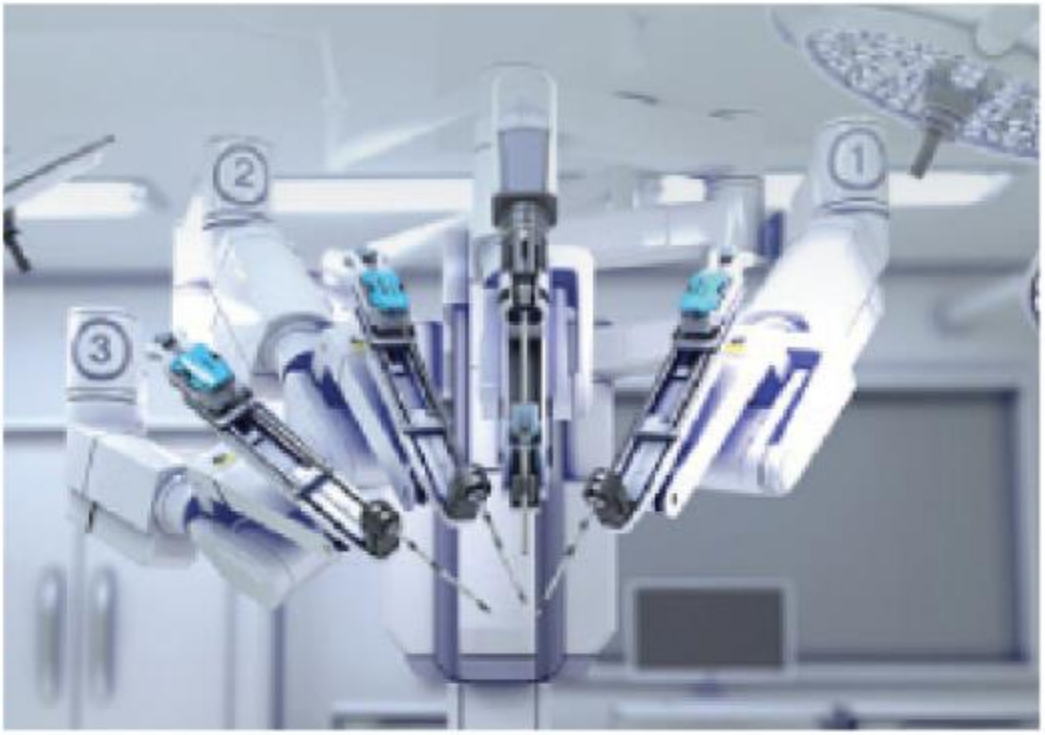
By acknowledging our limitations, focusing on real needs, and collaborating, we can make significant strides in transforming healthcare with AI.

The event was the perfect occasion to share our passion for data science and AI in healthcare. I am grateful for the lively discussions and the chance to connect with like-minded individuals. Let's continue to push the boundaries of what's possible with AI in healthcare.

Adriyana was featured as Woman in Computer Vision in the August 2019 issue of Computer Vision News







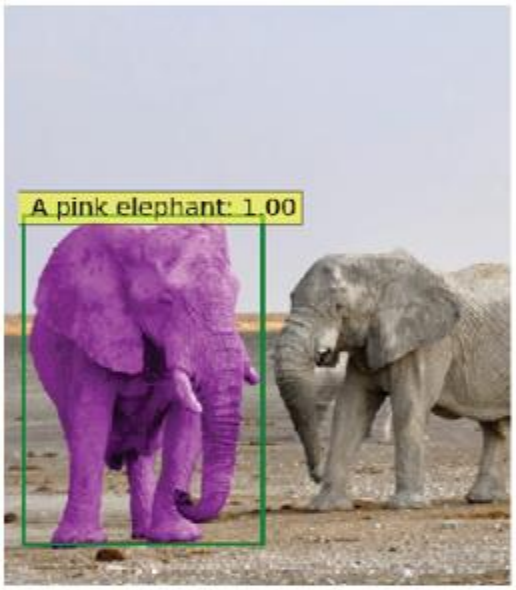
IMPROVE YOUR VISION WITH

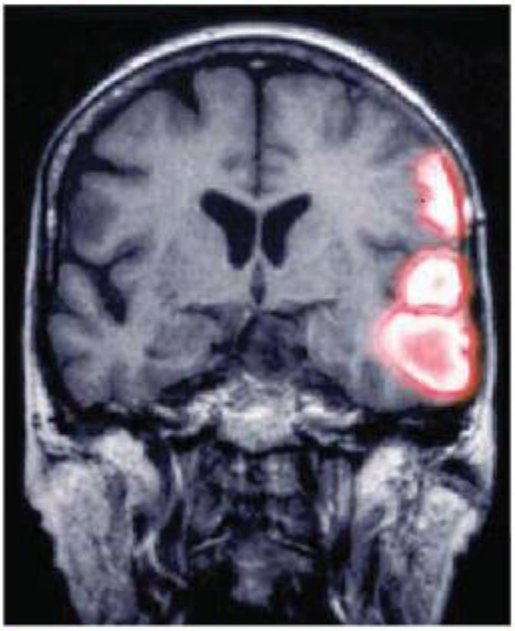
Computer Vision News

SUBSCRIBE

to the magazine of the algorithm community

and get also the new supplement Medical Imaging News!





RSIP VISION