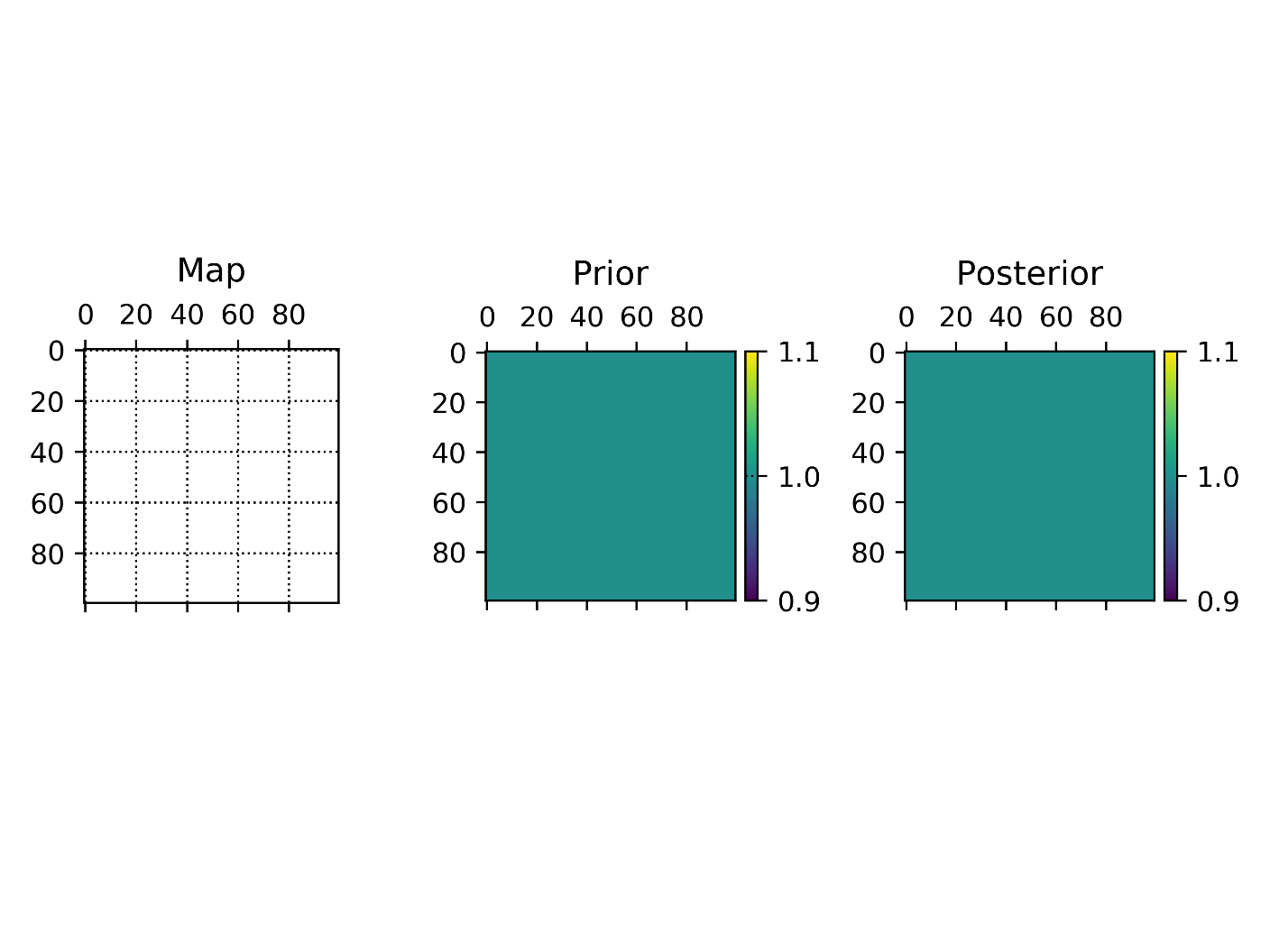
Aufgabe 2.1 - Markov Lokalisierung

Markus Mennel

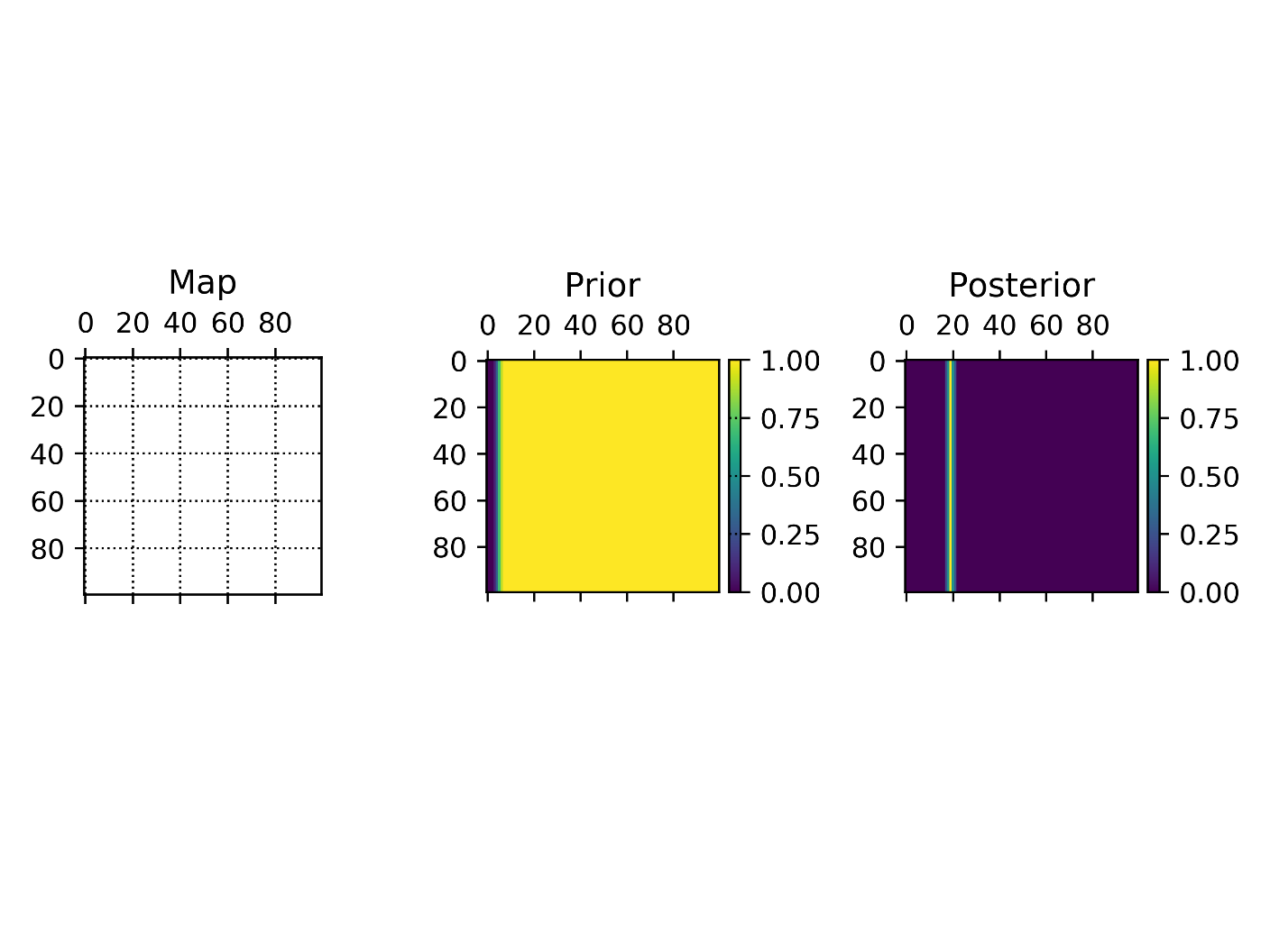
Ein Roboter soll sich in einer Karte mittels Markov-Lokalisierung bewegen. Es wird nur eine Unsicherheit der Position in x und y angenommen. Die Winkelausrichtung des Roboters erfolgt exakt. Als Karte wird ein 2D Occupancy Grid mit 100 x 100 Zellen verwendet. Im Occupancy Grid wird die Wahrscheinlichkeit, dass eine Zelle vom Roboter besetzt ist zwischen 0 und 1 eingetragen.

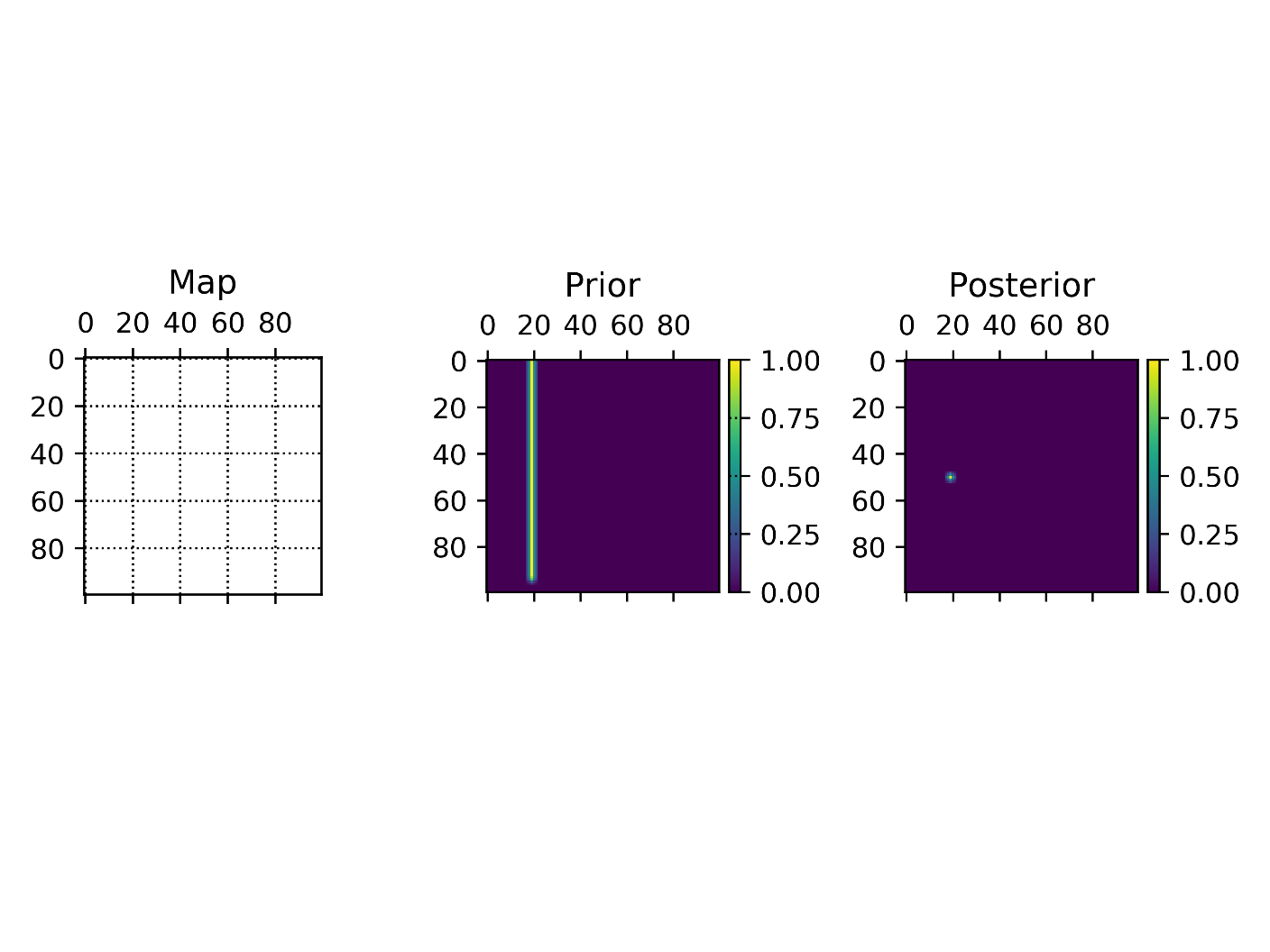
# Teil 1:

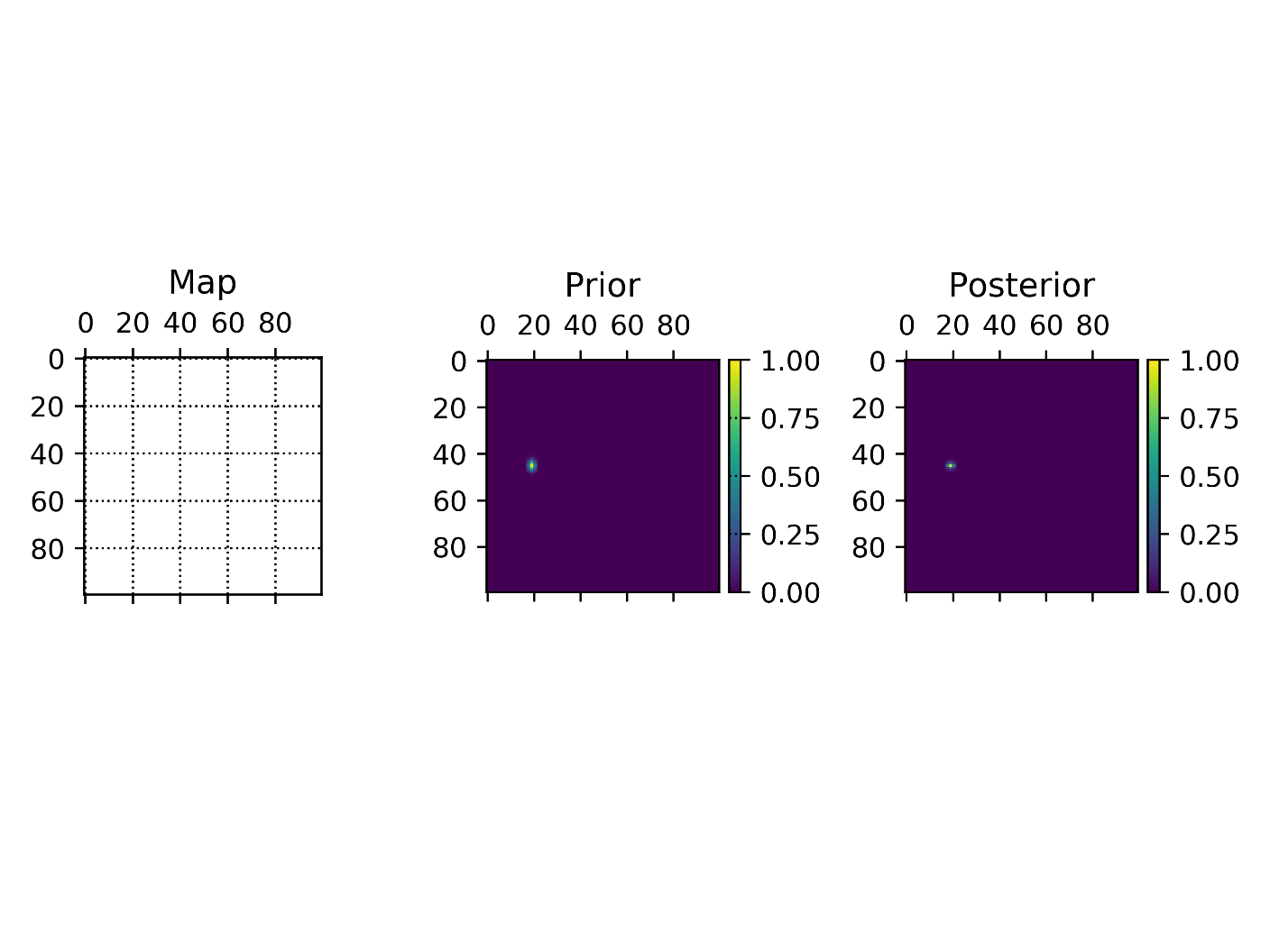
Ausgangssituation:



1. Schritt nach **rechts**, Sensor misst einen Abstand von **80 Kästchen**



1. Schritt nach **oben**, Sensor misst einen Abstand von **50 Kästchen**.  
   
2. Schritt nach **oben**, Sensor misst einen Abstand von **45 Kästchen**.



Nach dem letzten Schritt steht in der Zelle (x=45, y=20) die höchste Wahrscheinlichkeit für die Position des Roboters.

# Teil 2:

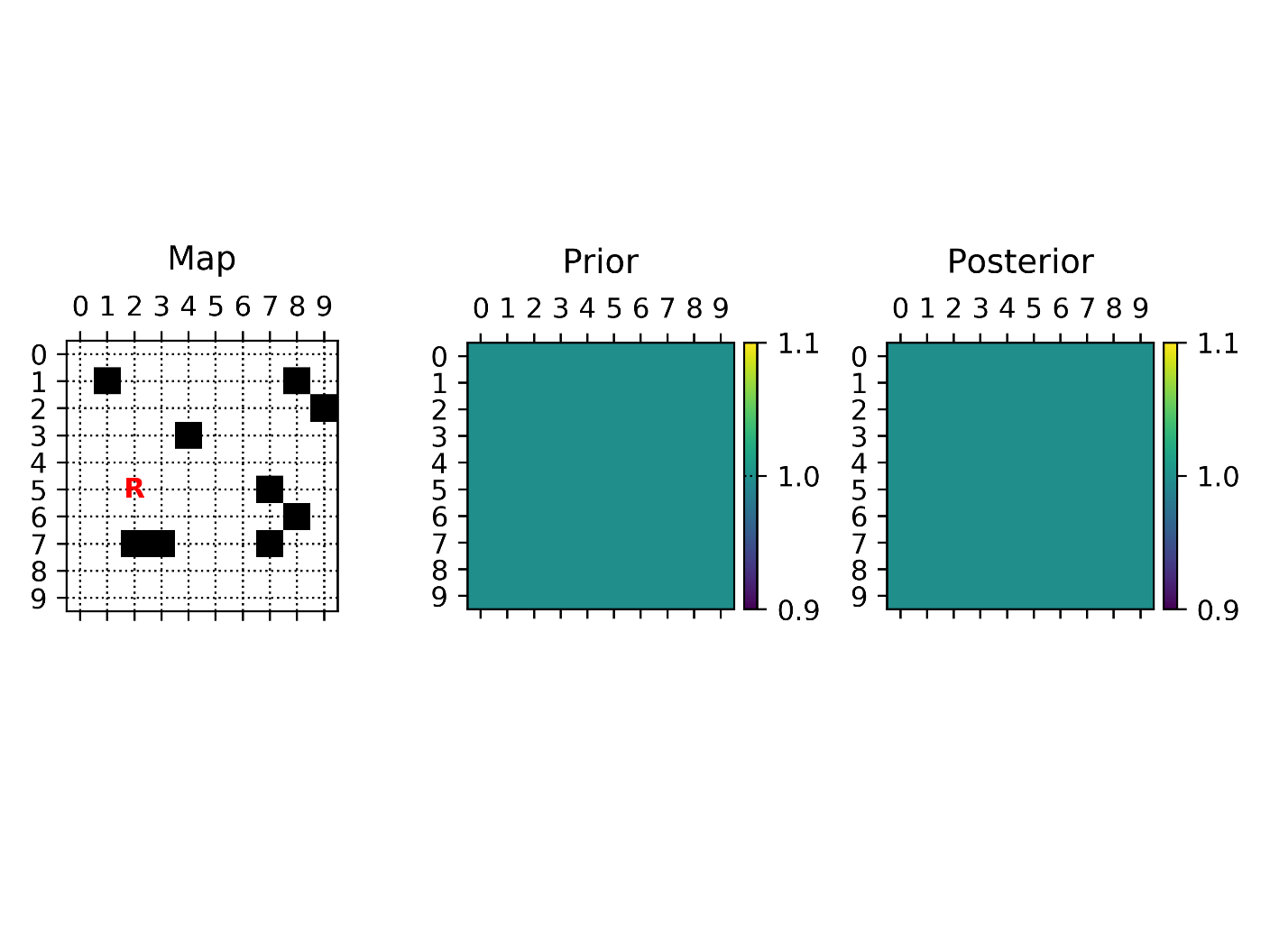
**Aufgabe:**

Inkludieren Sie die Möglichkeit, Wände (Hindernisse) in der Karte einzuzeichnen. Adaptieren Sie das Messmodell und die Berechnung der Prior und Posterior damit die Wände und Hindernisse richtig berücksichtigt werden können. Fügen Sie einige Wände ein und demonstrieren Sie das Kidnapped Robot Problem.

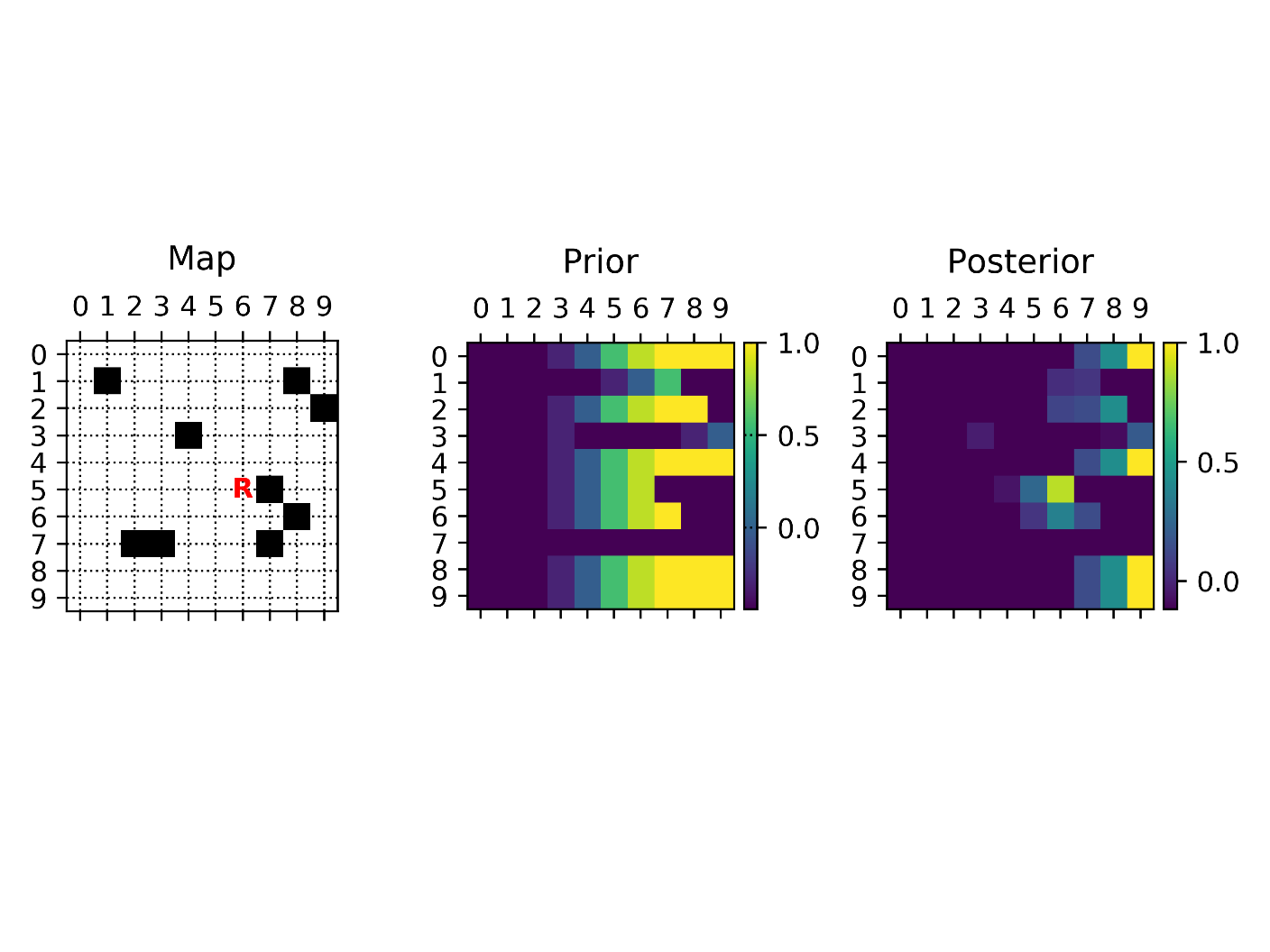
Die Karte wurde für diesen Teil auf eine Größe von 10x10 verkleinert, damit die Plots leichter mit freiem Auge lesbar sind. Natürlich funktioniert die Lösung auch für größere Grids.

Wände sind in der Karte als schwarze Kästchen dargestellt. Der Roboter erhält eine Matrix, in welcher Wände mit einer 1 eingetragen sind und Felder, auf welchen er sich frei bewegen kann mit einer 0 gekennzeichnet sind. Zu beginn befindet sich der Roboter an der Position (x=5, y=2). Jedoch kennt er natürlich seine wahre Position nicht. Die aktuelle Position des Roboters ist mit einem roten R in der Karte eingezeichnet.

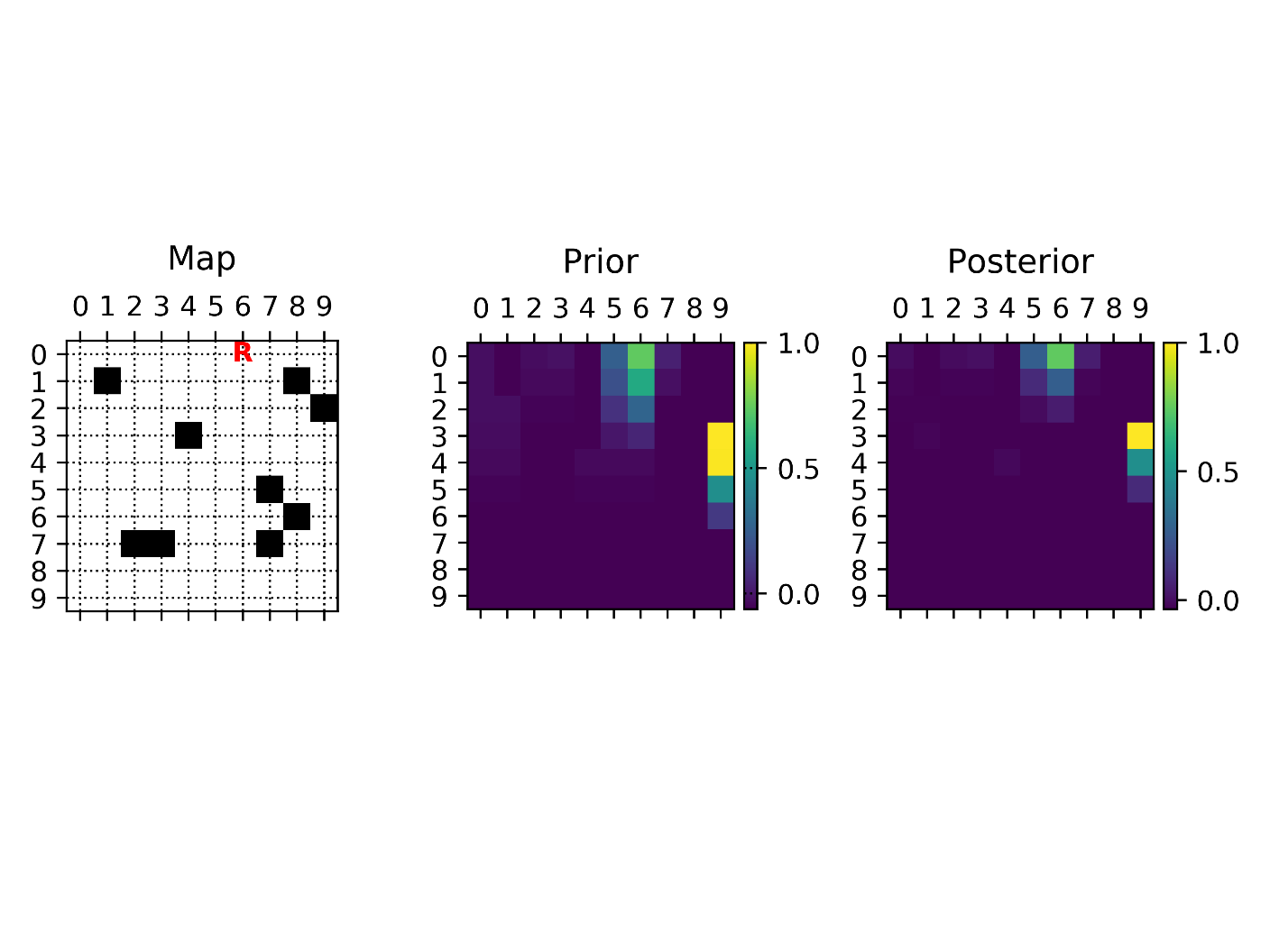
Die Ausgangssituation sieht wie folgt aus:

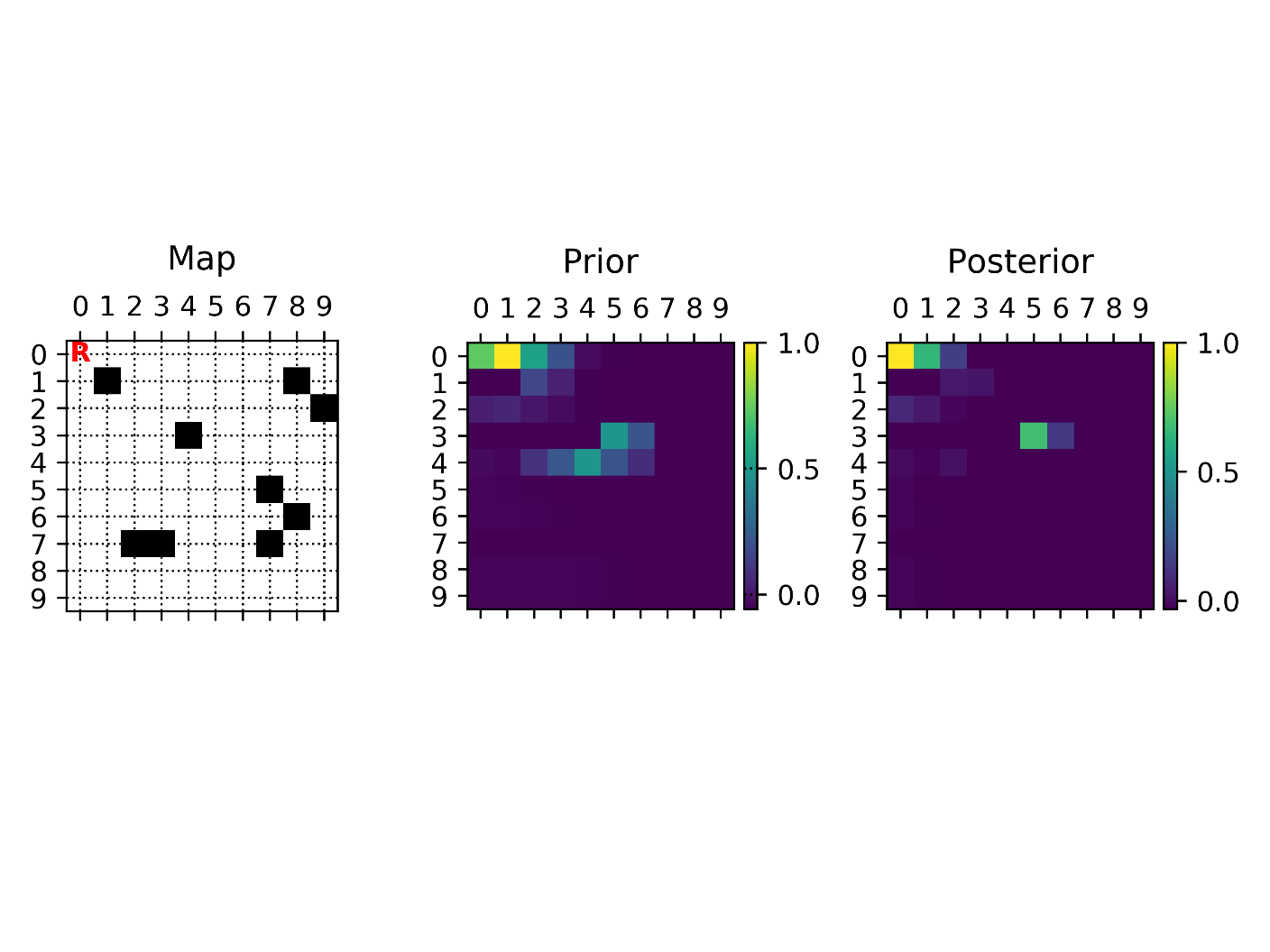


**1. Schritt:**   
Schritt nach **rechts**, Sensor misst einen Abstand von **0 Kästchen**.



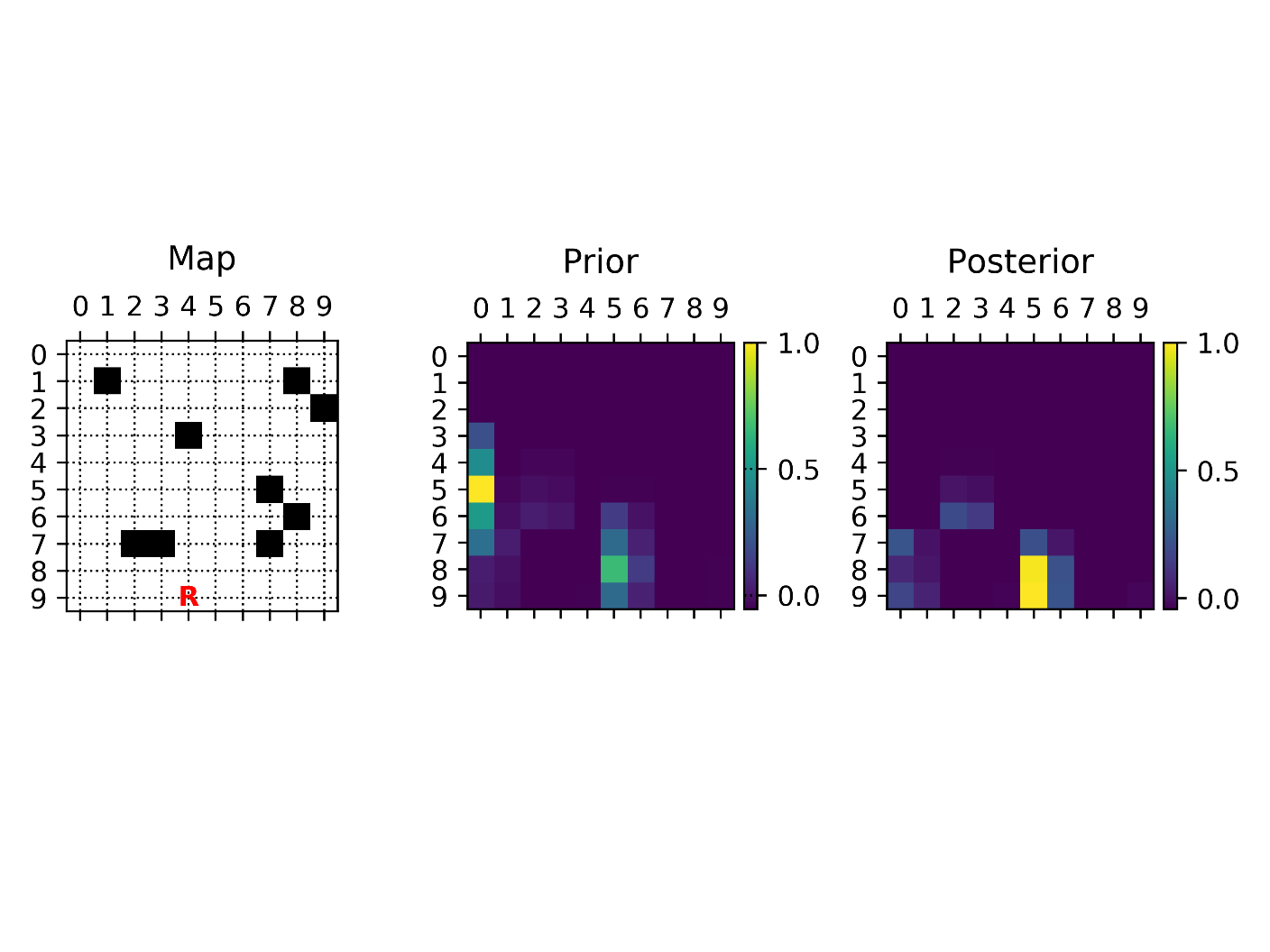
**2. Schritt:**   
Schritt nach **oben**, Sensor misst einen Abstand von **0 Kästchen**.



**3. Schritt:**  
Schritt nach **links**, Sensor misst einen Abstand von **0 Kästchen**.  
  


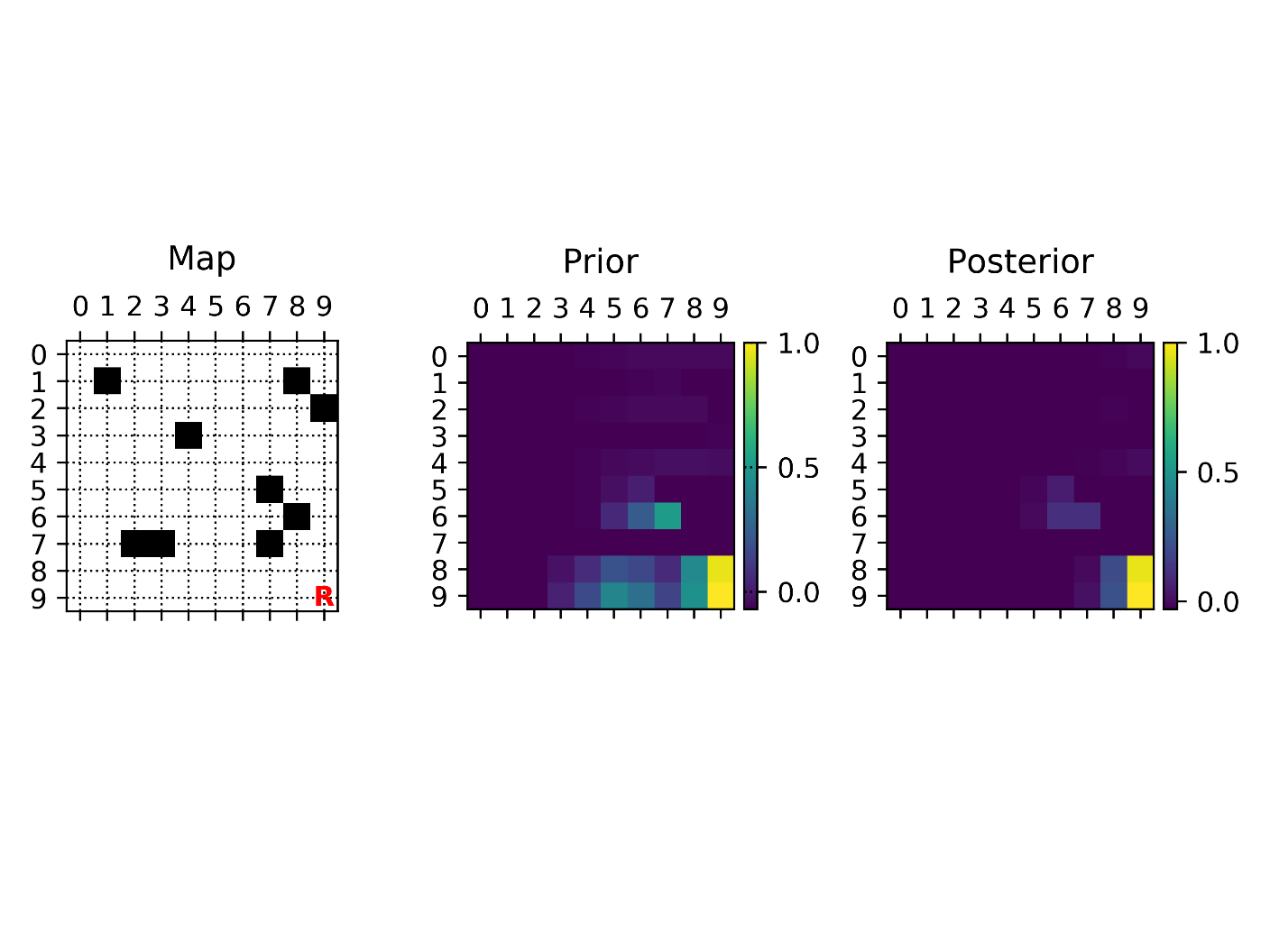
**4. Schritt:**   
**Vor diesem Schritt wurde der Roboter gekidnapped und auf die Position (x=6, y=4) gesetzt.**

Schritt nach **unten**, Sensor misst einen Abstand von **0 Kästchen**.



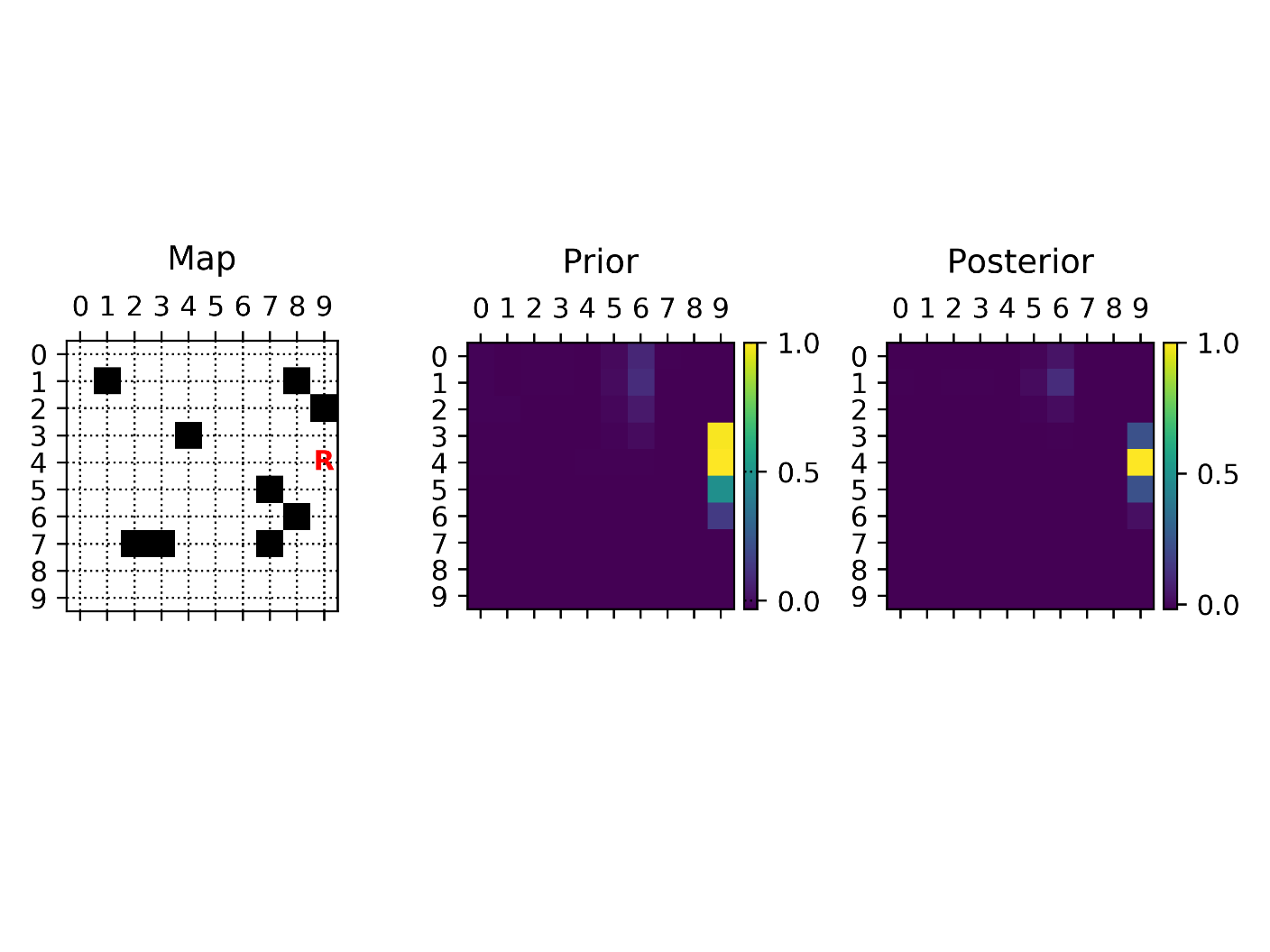
**5. Schritt:**

Schritt nach **rechts**, Sensor misst einen Abstand von **0 Kästchen**.



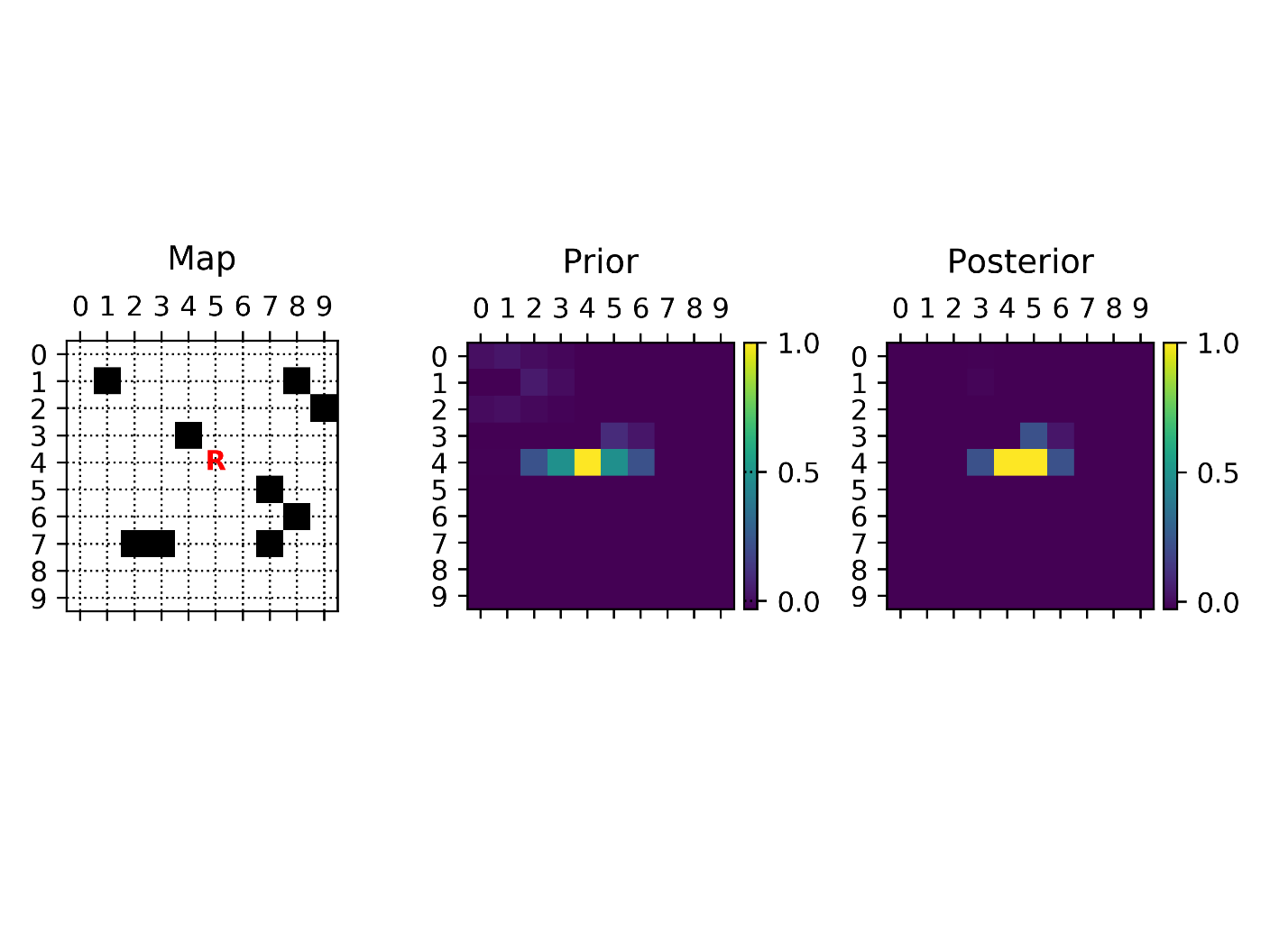
**6. Schritt:**

Schritt nach **oben**, Sensor misst einen Abstand von **1 Kästchen**.



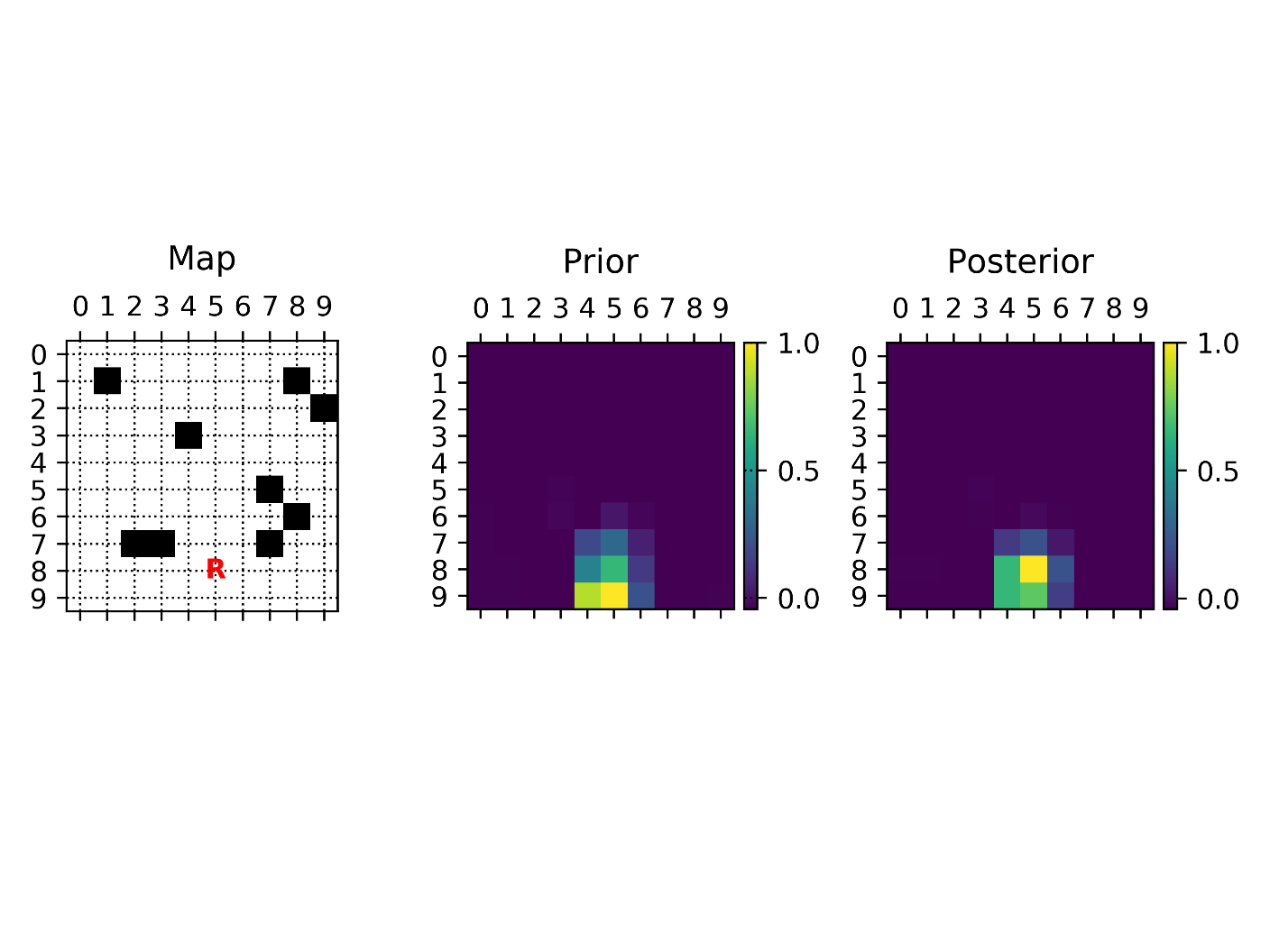
**7. Schritt:**

Schritt nach **links**, Sensor misst einen Abstand von **5 Kästchen**.



**8. Schritt:**

Schritt nach **unten**, Sensor misst einen Abstand von **1 Kästchen**.



Trotz kidnapping hat der Roboter sehr schnell seine neue Position erkannt und weiß, dass er im letzten Schritt auf Position (x=8, y=5) stehen muss.